



Finanziato
dall'Unione europea
NextGenerationEU



Ministero
dell'Università
e della Ricerca



Italiadomani
PIANO NAZIONALE
DI RIPRESA E RESILIENZA



UNIVERSITÀ
DEGLI STUDI DI BARI
ALDO MORO

UNIVERSITA' DEGLI STUDI DI BARI ALDO MORO

Dipartimento di Ricerca e Innovazione Umanistica

DOTTORATO DI RICERCA IN PATRIMONI STORICI E FILOSOFICI PER UNA INNOVAZIONE
SOSTENIBILE

CICLO XXXVIII

Settore Scientifico Disciplinare PHIL-02/B

*Digital Design come rapporto tra l'essere umano e le forme della computazione. Una
storia critica della tecnologia*

Dottorando: Dott. Pasquale Grieco

Supervisore: Ch.mo Prof. Francesco Paolo de Ceglia

Coordinatore: Ch.mo Prof. Francesco Paolo de Ceglia

ESAME FINALE 2026

*Finanziato dall'Unione Europea - Next Generation EU, Missione 4 Componente 2 CUP H91I22000450007 Investimento 3.3
D.M. n. 352 del 9 Aprile 2022 – Piano Nazionale di Ripresa e Resilienza. Progetto “Machine learning in una società
inclusiva: fondamenti epistemologici e sfide etiche” cofinanziato da Exprivia Spa*

RINGRAZIAMENTI

Dopo tre anni, il sogno di un bambino che negli anni Novanta leggeva affascinato *Jurassic Park* e immaginava di aggiungere due lettere e un punto prima del suo nome – “Dr.” – sta per diventare la realtà di un quarantenne. È pressoché impossibile ringraziare in modo appropriato tutte le persone straordinarie senza le quali questo traguardo sarebbe rimasto, con ogni probabilità, soltanto un sogno.

In primo luogo, tutta la mia gratitudine va al mio tutor, il Prof. Francesco Paolo De Ceglia, che con la sua guida costante – a volte ferma, talvolta persino aspra, ma sempre umanissima – ha saputo trarre il meglio da me. Grazie, Professore, per aver deciso di scommettere su di me quando io stesso non lo avrei fatto.

Un ringraziamento speciale va anche alla co-tutor di questo progetto, la Prof.ssa Annalisa Caputo, per la pazienza dimostrata di fronte alle mie innumerevoli mancanze.

Un sentito grazie va al Prof. Francesco Marrone, per il suo supporto scientifico, umano e personale, e per la fiducia che mi ha sempre accordato. Professore, se ho scalato qualche gradino nell'*ordo cognoscendi*, è anche merito suo.

Al Prof. Edoardo Datteri vanno tutta la mia gratitudine e la mia stima. Non troverò mai parole adeguate per esprimere quanto gli sia riconoscente per aver mostrato - a me, che ero già un “fan” dei suoi lavori scientifici ben prima di avere il privilegio di conoscerlo personalmente e di apprezzarne la sua immensa statura umana - stima per il mio lavoro, per le mie potenzialità e per la mia persona. Senza la sua apertura di credito “alla cieca” nei miei confronti, non avrei mai realizzato molte delle cose di cui vado più fiero in questo percorso. Non vedo l’ora di continuare a girare la Puglia e condividere con lui banchetti di specialità locali, mentre parliamo di robotica, cibernetica, storia locale e se esistano o meno le rappresentazioni.

Il mio cuore è rimasto anche nella mia seconda patria accademica, Nijmegen: non trovo parole per descrivere l’ammirazione e l’affetto che provo per la Prof.ssa Carla Rita Palmerino e per il Prof. Christoph Lüthy. Che persone della vostra statura umana e scientifica mi abbiano accolto, fatto sentire a casa e permesso di vivere, per mesi, la realtà di ciò che avevo sempre sognato, è un debito impagabile che porto con me. Essere l’attuale detentore del titolo di “capocannoniere” del *Center for the History of Philosophy and Science* è uno dei riconoscimenti di cui vado più fiero. A tutto il gruppo di Nijmegen - Gabriel, Maria, Antonio, Graziana, il sommo maestro dell’esistenza Osvaldo e tutti gli altri - va il mio abbraccio e il mio affetto, grande quanto i pacchi che arrivavano da giù! Non c’è nebbia che possa offuscare il ricordo della vostra accoglienza.

Un ringraziamento “internazionale” speciale va al Dr. Jaap-Henk Hoepman, dell’iHub – *Interdisciplinary Research Hub on Digitalization and Society*, e alla Dr.ssa Nina Poth, del *Center for Cognition, Culture, and Language* (CCCL) della Radboud University di Nijmegen, per avermi accolto con pazienza e umanità nei lavori dei loro centri di ricerca, avermi mostrato vie di indagine altrimenti per me impensabili e per aver confermato la bontà di alcune delle idee presentate in questa tesi.

Al Prof. Corrado Mencar va la mia profonda gratitudine: la sua speciale umanità e la sua competenza tecnica mi hanno onorato della possibilità di collaborare con lui e di poterlo considerare un amico.

Amici, oltre che maestri e guide, sono stati anche il Dott. Roberto Anglani e il Dott. Vitantonio Fascilla. Del primo porterò sempre con me l’entusiasmo contagioso, la passione “filosofica”, la competenza scientifica e il cuore enorme - qualità che sono state per me uno stimolo costante e un’ispirazione continua. Del secondo, amico e preziosissimo maestro, non smetterò di essere grato per avermi mostrato, con pazienza e stupore, come le mie elucubrazioni teoriche potessero prendere vita concreta, facendomi sentire parte della sua squadra come un membro di famiglia.

Un ringraziamento sentito anche al Dott. Nicola Tino, al Dott. Domenico Favuzzi di Exprivia e a tutta Exprivia S.p.A. per aver deciso, *against all odds*, di investire su di me. Anche nel business, Vitantonio, “si può prenderla con filosofia”!

Un posto speciale in questi ringraziamenti lo hanno tutti i colleghi e i membri del Seminario di Storia della Scienza, che sono stati compagni, complici, mentori e guide. Francesca ed Eleonora, le mie due sorelle maggiori: mi sarei arreso un milione di volte se voi non foste state le persone straordinarie che siete. Fabio Lusito, Fabio Frisino, Andrea Maraschi e Stefano Daniele, fratelli maggiori, amici e “Virgili” in questo viaggio bellissimo, spericolato e a volte spaventoso. Simona, *partner in crime* di mille follie accademiche, e Virginia e Angelica, amiche e colleghe preziosissime: nessuna ansia da dottorato potrà mai farmi rimpiangere di avervi conosciute.

A Ivano va un ringraziamento particolare. Non saprò mai ripagarlo per essermi stato vicino, per avermi coinvolto in mille progetti e collaborazioni, per essere un essere umano speciale. La sua solidità, pazienza e intelligenza fuori da ogni descrizione sono state l’elemento di realtà che ha reso possibili le mie “donchisciotterie”. Niente di ciò che ho realizzato sarebbe stato possibile senza di te, Ivano: sei il “dottor fratello minore” che ho sempre desiderato.

Un abbraccio e un ringraziamento altrettanto grande vanno a tutti i “fratellini” e le “sorelline” del Seminario e del corso di dottorato: Iolanda, Antonella, Roberta, Ilaria, Sara,

Alessio, Alessandro e tutti gli altri colleghi. Spero di essere stato per voi uno “zio cool” degno anche solo di un milionesimo di quanto preziosi siete stati per me.

Al Prof. Liborio Dibattista va un ringraziamento speciale: è stato, segretamente, il mio modello di insegnante, ricercatore e scienziato. Il suo stile e il suo approccio alla scienza e all'accademia sono stati per me una costante ispirazione. Lunga vita al *Graviner Kreis*!

Un ringraziamento va anche alla direzione e al coordinamento del corso di dottorato. Gestire un “mare in tempesta” di questo calibro non deve essere stato affatto facile, e se questo sogno prende corpo è anche grazie al duro e silenzioso lavoro di tutte le persone coinvolte. E un abbraccio speciale va alla mia *angstmaatje* Letizia: se siamo sopravvissuti al *broodje haring*, sopravviveremo anche al Tarantino!

Porto con me un enorme debito di gratitudine verso le mani tese di persone meravigliose come Alberto e Paola, sempre pronte a recuperarmi e a indicarmi la via d'uscita dai tunnel - spesso immaginari - in cui immancabilmente finisco per perdermi.

Non posso che chiudere ringraziando la mia famiglia per il supporto, la vicinanza e l'affetto - e soprattutto per la pazienza - dimostrata in questi tre anni, mentre inseguivo un sogno, abbandonando per un momento il “posto fisso”. Grazie mamma, papà, Silvio e Viviana: senza di voi non sarei nulla.

E meno di nulla sarei senza la mia compagna, Francesca. Mi ha tenuto per mano e mi ha mostrato la via ogni volta che cadevo; la sua sola presenza era uno stimolo a rialzarmi e a lottare. Ti amo: quelle due lettere e un punto che posso finalmente mettere prima del mio nome servono anche a sentirmi, almeno un po', all'altezza del tuo amore.

だってばよ！

Indice

Introduzione.....pp. 1-20

Cap. 1: Understanding Whole System: Intorno alle radici scientifico-culturali del rapporto uomo-macchina. Dall'archeologia cibernetica all'idea di personalizzazione

1. **Scarti della macchina e poesie d'amore elettronico**
 - a. Sono uno studente: non piegare, infilzare o mutilare.....pp. 21-26
 - b. The Combine strikes again.....pp. 26-29
2. **Oggi il predittore, domani il mondo: genesi ed eredità della visione del mondo cibernetica**
 - a. Macchine perturbanti.....pp. 29-39
 - b. Questione di feedback.....pp. 39-50
 - c. Il seme cibernetico nell'era dell'informazione.....pp.50-66
3. **Rumore, entropia, informazione: la cosa migliore dai tempi degli psichedelici**
 - a. Tabula logica elettrificata: pericoli matematici, statistica e relè.....pp. 66-79
 - b. Geni algebrici, voci sintetizzate.....pp. 79-88
 - c. Segretezza, ridondanza, entropia: crittoanalisi e teoria dell'informazione.....pp. 88-100
 - d. "Fa' come Wiener": aggiusta il tiro. Cervelli informazionali e liberazione individuale.....pp. 101-108

Cap. 2: Dalla carne al calcolo e ritorno: Computazioni; Imitazioni; Implementazioni

1. **Formalità: *Ignorabimus quia Incompletum*.....pp. 109-120**
2. **Algoritmi dell'indecisione.....pp.120-130**
3. **Porridge freddo computazionale.....pp. 131-147**
4. **Imitare quel ronzio nella mia testa.....pp. 147-162**
5. **Scacco matto: un matematico tra simboli, neuroni e calcolatori.....pp.163-178**

Cap. 3: Il visibile e l'invisibile

1. **Istruzioni storiche per un cervello digitale.....pp. 179-189**
2. **Bush e Wiener a caccia dell'equazione differenziale.....pp. 189-198**
3. **Più forte delle bombe: il superamento dell'analogico e l'informazione digitale.....pp. 199-214**

Cap. 4: Il mondo in un nastro: Macchine universali, calcolatori seriali

1. **HQIBPEXEMUG, o di come uno sconosciuto gentiluomo tedesco diede avvio alla Rivoluzione dei Computer.....pp. 215-233**
2. **Colossi, valvole e profeti: le vie del digitale.....pp. 234-264**
3. **Turing in hardware: ACE e la Macchina di Manchester.....pp.264-291**
4. **Un avvento cibernetico: EDVAC e il suo design.....pp. 291-308**

Cap. 5: Diffusioni digitali: Serialità, elettronica, società

1. ***Rien ne va plus*: la nascita della simulazione.....pp. 309-331**
2. **Liberi dalla paura, liberi dalla matematica: mainframes e programmazione.....pp.331-353**
3. ***California dreamin'*: un viaggio elettronico verso la Silicon Valley.....pp. 354-374**
4. **Una società omeostatica, una liberazione cibernetica.....pp.375-385**
5. **Un universo di informazioni: dati e interfacce.....pp.385-399**

Cap. 6: "Johnny Builds Bombs and Johnny Builds Brains": Verso la struttura computazionale del sistema nervoso

1. **Microorganismi nell'ENIAC: la cibernetica riformata**.....pp. 400-410
2. **Una logica degli errori: dagli automi alla vita**.....pp. 410-440
3. **L'irragionevole complessità del cervello**.....pp. 441-458

Cap.7: Connessioni: Reti neurali, prospettive critiche, design

1. **Dall'automa cerebrale alle reti: avventure e disavventure del Percettrone**.....pp. 465-473
2. **Un principe azzurro subsimbolico: backpropagation, parallelismo e macchine probabilistiche**.....pp. 473-483
3. ***Going deep*: scenari tecnici e critici dell'IA contemporanea**.....pp.483-501
4. ***Design for Action*: verso un'etica progettuale**.....pp. 502-514

Digital Design: *Un framework etico per la progettazione di tecnologie AI-HCI come soluzione sostenibile alle sfide della complessità e dell'innovazione*.....pp.515-609

Conclusioni: Un ecosistema di informazioni, il design per un "uso umano degli esseri umani" pp.610-632

Bibliografia.....pp.630-674

Introduzione

Tra tecnologia e storia: una domanda e un metodo

Il presente lavoro prende avvio da un nucleo metodologico e tematico radicato nella Storia della scienza e delle tecniche (PHIL-02/B), intesa come analisi critica dei contesti storici, concettuali, culturali e materiali entro cui le tecnologie computazionali si sono sviluppate. A partire da questo impianto, la ricerca si apre al confronto con altri ambiti di indagine - in particolare la filosofia della scienza, la filosofia della mente, le scienze cognitive, la computer science e la teoria della computazione, nonché gli studi sull'interazione uomo-macchina e il design tecnologico - secondo un'articolazione che è dettata dall'oggetto stesso di indagine.

Le tecnologie prese in esame, infatti, non possono essere comprese adeguatamente entro i confini di un singolo dominio disciplinare: esse sono al tempo stesso artefatti tecnici, modelli teorici, strumenti cognitivi e dispositivi operativi che intervengono nei processi di conoscenza e di azione.

La contemporaneità che abitiamo è sempre più attraversata dalle *tecnologie dell'informazione e della comunicazione* (ICT) digitali, divenute ormai imprescindibili costanti della nostra esistenza quotidiana. In pochi decenni, gli strumenti digitali si sono fatti più piccoli, più leggeri, più intimi, più personalizzati: dal calcolatore da tavolo al computer portatile allo smartphone, fino ai dispositivi indossabili, agli assistenti virtuali, ai chatbot e alle applicazioni dell'Intelligenza Artificiale Generativa. La prova è nel palmo della mano di ciascuno, dove un piccolo artefatto digitale è la porta ad un universo di informazioni.

Questa interazione sempre più *simbiotica* che ridefinisce l'immagine dell'umano e del rapporto con la macchina, e qui risiede la tesi di fondo che nel lavoro seguente sarà messa alla prova, è frutto di un *farsi storico*: solo determinate contingenze storiche e determinati contesti culturali hanno reso possibile lo strutturarsi di questa relazione e quest'ultima diventa comprensibile solo alla luce di questa storicità. Ciò rende non solo possibile, ma necessario l'impiego degli strumenti analitici della storia della scienza e della riflessione filosofica per analizzarne caratteristiche, direttive, potenzialità e criticità. La sfida di questo lavoro è mostrare come l'impiego di fonti storiche dirette - lettere, rapporti tecnici, documentazione d'archivio e testimonianze di prima mano - e indirette, affiancate da una ricostruzione concettuale dei presupposti teorici che hanno orientato lo sviluppo delle tecnologie computazionali, possa essere utile nei luoghi dove le tecnologie in questione vengono

effettivamente implementate per orientarle verso la sostenibilità e la valorizzazione dell'essere umano che con esse si relaziona.

La domanda che struttura la ricerca è dunque la seguente: in che modo una ricostruzione storica della relazione tra l'essere umano e le tecnologie digitali dell'informazione e della comunicazione consente di individuare le direttrici concettuali che ne hanno orientato lo sviluppo, e di assumerle come base per un'analisi critica delle forme contemporanee dell'interazione uomo-macchina e delle loro modalità di implementazione? In altri termini, lo sviluppo di concetti quali feedback, informazione, digitale, controllo e automazione — considerati non come esiti di un determinismo tecnologico, ma come prodotti di dinamiche socio-tecniche storicamente situate — può essere ricostruito, attraverso il lavoro teorico e tecnico di figure come Wiener, Turing, Shannon e von Neumann, e impiegato come risorsa operativa nei contesti contemporanei di progettazione e implementazione delle tecnologie digitali?

A tal fine, il lavoro intende individuare alcuni concetti-chiave che fungono da prisma interpretativo nella storia delle dinamiche del rapporto tra uomo e macchina, naturale e artificiale, carbonio e silicio, dai primi sviluppi della computazione e della cibernetica bellica fino all'Intelligenza Artificiale generativa contemporanea. L'obiettivo è mostrare come tali concetti possano fornire indicazioni pratiche – il presente lavoro, anche nell'ambito della dimensione aziendale del progetto, ne offrirà una possibile concretizzazione attraverso un framework per la progettazione eticamente orientata di sistemi di intelligenza artificiale e per il design delle interfacce uomo-macchina (HCI) – per l'implementazione odierna di tecnologie capaci di far tesoro tanto dei guadagni quanto delle criticità emerse da un'analisi storica e critica di questo rapporto, delle sue caratteristiche e delle questioni che esso solleva.

Io o 1/0?

La diffusione e la pervasività delle ICT fanno sì che oggi ogni gesto, ogni movimento, ogni interazione è oggi mediata - o meglio, clusterizzata in dati - tracciata, codificata, interpretata da qualche forma di tecnologia digitale: un algoritmo di raccomandazione che anticipa i nostri desideri; un sensore biometrico che registra il battito del cuore; un sistema di navigazione che apprende le nostre abitudini di spostamento. Queste tecnologie non si limitano ad ampliare le capacità operative dell'uomo: esse penetrano nella sua architettura cognitiva [Clark, 2025], integrandosi nei processi di percezione, decisione e memoria. Il digitale sta operando, a velocità sempre maggiore, una re-ontologizzazione e una re-epistemologizzazione di categorie che

davamo acquisite come lascito della modernità: uomo, mondo, sé, sono sempre più considerabili e *trattabili* come entità *informazionali* [Floridi, 2022], come pattern strutturati e computabili di informazioni, in cui il confine è soltanto analitico, non più ontologico.

Nel discorso pubblico e nei media, termini fino a poco tempo fa appannaggio di un ristretto novero di specialisti, accademici e tecnologi - *cyber, big data, informazione, machine learning, deep learning, Large Language Model* - sono diventati parte del linguaggio comune, penetrando nel lessico (e nell'agire) politico, economico e perfino quotidiano. Questi plasmano oggi una narrazione che oscilla tra aspettative messianiche, che vedono nella tecnologia la chiave di una nuova era di progresso e liberazione, e visioni *disruptive* - come ad esempio scenari di disoccupazione di massa - e apocalittico/catastrofiche, in cui gli stessi strumenti digitali appaiono come una potenza senza controllo, capace non solo di trasformare l'attività umana, ma di sostituire - o addirittura sottomettere - l'uomo alle proprie logiche.

In questo immaginario ambivalente, la tecnologia digitale si configura insieme come strumento e come destino, come protesi della nostra intelligenza e come minaccia alla nostra autonomia.

Dietro questa enorme massa di bit - dai pochi byte di memoria dell'ENIAC, meno di quanto oggi occupi un singolo carattere di una foto scattata da uno smartphone, agli oltre 200 zettabyte che costituiscono l'attuale ecosistema digitale globale (una quantità pari a circa 44.600 miliardi di DVD, che impilati formerebbero una colonna alta oltre 53 milioni di chilometri, sufficiente a coprire la distanza tra la Terra e la Luna per più di 130 volte) - si cela la storia dell'espansione di un nuovo mondo, quello digitale, detonato con la stessa forza di una tecnologia che ha contribuito a plasmarlo: quella degli ordigni atomici.

Ma dietro gli 0 e gli 1 (gli ingredienti minimali del digitale) si staglia, sempre più sfocata, l'immagine dell'essere umano, in una dinamica progressiva di *imitazione - riduzione-integrazione/blurring* umano/macchina - *riproduzione/reificazione* in dati e previsione mediante algoritmi [Numerico, 2021].

Questa è la dinamica che segna il solco dell'*Überfrage* che accompagna l'intero svolgersi di questo lavoro. Un percorso che, tra mondo analogico e virtuale, prende avvio nell'assoluta estate parigina del 1900 - quando, al secondo Congresso Mondiale dei Matematici, si gettarono le fondamenta della formalizzazione moderna - e si prolunga idealmente fino all'Aula Magna dell'Archiginnasio di Bologna, per poi snodarsi attraverso luoghi simbolici come l'Inghilterra di Bletchley Park, la Pennsylvania della Moore School, la California della Silicon Valley e il mondo digitale contemporaneo. Un itinerario innescato dalla scintilla delle simulazioni del metodo

Monte Carlo - concepite per i calcoli sulla bomba all'idrogeno - e culminato con la nascita di ARPANET, antesignana del moderno web.

Attraverso questi passaggi, la tesi si articola come una *mappa storica e concettuale del design dell'interazione uomo-macchina*: un tentativo di restituire la trama evolutiva con cui, nel corso del XX secolo, i due poli di questa relazione - l'essere umano e la macchina che egli progetta - si sono progressivamente trasformati e ridefiniti a vicenda.

Un dialogo che, lungo tutto il XX secolo, è sorto in un preciso contesto storico-culturale, quello del secondo conflitto mondiale, che si raccoglie in una domanda fondamentale: "Stiamo cercando di creare un *sostituto* meccanico dell'essere umano, o uno strumento che *lavori in simbiosi* con l'uomo, un'estensione delle sue facoltà mentali?" [Ceruzzi, 2012, p. 42, trad. mia enfasi mia]¹.

In questo orizzonte, il lavoro che qui si presenta ha cercato di mostrare come concetti, direttrici di sviluppo e problemi che ancora oggi segnano il rapporto fra essere umano e tecnologia siano sorti fin dalle origini del pensiero cibernetico e computazionale. Figure, che diventano sempre più conosciute al grande pubblico, come Norbert Wiener, Claude Shannon, Alan Turing e John von Neumann, accanto ad altri protagonisti meno noti ma altrettanto decisivi – una galleria che comprende, tra gli altri, figure come Thomas Flowers, Vannevar Bush, John Mauchly, J. Presper Eckert, Herman H. Goldstine, Grace Hopper – hanno contribuito a delineare un nuovo immaginario scientifico e tecnico dell'uomo e della macchina. A questi si affiancano i nomi quasi cancellati della storia: le WRENS, le operatrici *della Women's Royal Naval Service* che a Bletchley Park gestivano le macchine Heath Robinson e i Colossus, e le sei programmatrici dell'ENIAC – Kay McNulty, Betty Jennings, Betty Snyder, Marlyn Wescoff, Fran Bilas e Ruth Lichterman – prime a tradurre, attraverso intelligenza e duro lavoro, in gesti e procedure concrete la logica delle macchine.

La ricostruzione che segue assume che nessuna riflessione critica sulle tecnologie computazionali e digitali, sulle loro possibilità e sui loro limiti, possa prescindere da una base storico-scientifica - costruita analizzando documenti, report tecnici e testimonianze di prima mano, sempre accompagnate da una riflessione concettuale - dotata di concreta competenza tecnica e matematica, senza la quale ogni discorso "critico" si condanna alla sterilità.

Gli artefatti digitali vengono qui interpretati, secondo una prospettiva che rifiuta il determinismo tecnologico, come artefatti sociotecnici, "combinazioni di cose e persone"

¹ "Do we construct machines that do what is technically feasible and adapt the human to their capabilities, or do we consider what humans cannot do well and try to construct machines that address those deficiencies?"

[Johnson, Miller, 2009, pp. 16-17, trad. mia]², nati da intrecci di conoscenze, istituzioni e valori. Proprio questo radicamento genealogico consente di comprendere come alcune direttrici di sviluppo abbiano potuto consolidarsi nel tempo e come, oggi, esse possano offrire strumenti per un'incorporazione consapevole dei valori dell'innovazione sostenibile nei luoghi in cui le tecnologie vengono progettate, implementate e messe in relazione con l'essere umano, condividendone ambienti, compiti, responsabilità e persino decisioni.

In questo senso, la tesi intende ricomporre – a diversi livelli d'astrazione, come in una conchiglia barocca, attraverso la storia delle macchine, dei loro creatori e delle *metafore* che questo rapporto ha generato e diffuso – la sagoma dell'essere umano che in esse si riflette e si trasforma.

Un seme cibernetico di informazione

A tal fine, come punto di vista prospettico, si è ritenuto particolarmente fecondo, seguendo l'approccio di neuroscienziati come M. A. Arbib, quella che, nel corso di questo lavoro, è stato più volte definita come "*metafora cibernetica*": l'analogia *funzionale* tra la dimensione biologica – rappresentata dal cervello umano e le sue facoltà – e la dimensione artificiale – incarnata/riprodotta nei diversi sviluppi degli strumenti computazionali, considerate entrambe come sistemi di elaborazione dell'informazione (*information processing*), e dunque come strutture analizzabili su una base concettuale comune.

Come ha notato lo storico della scienza R. Kline, "Il sostantivo *informazione* e il prefisso *cyber-* segnano il nuovo vocabolario del nostro tempo. Essi modellano il modo in cui parliamo, pensiamo e agiamo nel nostro presente e futuro digitale: dalle visioni utopiche evocate da espressioni come era dell'informazione e cyberspazio alle visioni distopiche associate a cyborg nemici e guerre cibernetiche. *Le tracce della cibernetica e della teoria dell'informazione permeano così le scienze, la tecnologia e la cultura della nostra vita quotidiana*" [Kline, 2015, p. 4, trad. mia enfasi mia]³.

² "STS scholars reject the idea that technology is material objects, and entreat us always to think of technology as sociotechnical systems (combinations of things and people)".

³ "The noun *information* and the prefix *cyber-* mark the new vocabulary of our time. They inform how we talk, think, and act on our digital present and future, from the utopian visions invoked by the terms *information age* and *cyberspace* to the dystopian visions associated with enemy *cyborgs* and *cyber warfare*. The traces of cybernetics and information theory thus permeate the sciences, technology, and culture of our daily lives"

L'analisi non poteva, allora, non prendere le mosse dall'*inception* della cibernetica: il *predittore antiaereo* [Wiener, 1956; Masani, 1990; Masani e Phillips, 1985, Conway, Siegelman, 2005], costruito da Norbert Wiener – una delle figure ricorrenti del nostro racconto – insieme all'ingegnere Julian Bigelow nell'ambito del *Project D.I.C. 5980 on Anti-Aircraft Predictor*, il suo “personale sforzo bellico”.

Concepito per prevedere e neutralizzare le traiettorie dei velivoli nazisti, quell'insieme di valvole, acciaio e ingranaggi divenne, come ha mostrato Galison [1994], un polo genetico di una teoria in cui si fuse la dimensione *cinestetica* e *propriocettiva* dell'elemento umano (il pilota) – clusterizzata in dati – con l'analisi statistica, la predizione automatica e il metallo della macchina: un sistema integrato, in cui la distinzione tra artigliere, arma e obiettivo sfumava fino a dissolversi.

Da quell'ammasso di acciaio e circuiti, costruito per abbattere il nemico, emerse una *nuova visione dell'uomo*: sepolto sotto strati e strati d'acciaio, dai quali era funzionalmente indistinguibile, concepito come nodo di un circuito retroattivo, un mero meccanismo omeostatico di *feedback*, di retroazione correttiva rispetto alle risposte ambientali, prevedibile e quantificabile. Da qui, Wiener avrebbe generalizzato questa intuizione fino a farne il principio di una teoria universale dell'organismo e della macchina, un'interpretazione del vivente e dell'artificiale come istanze studiabili sotto una base comune.

Eppure, dietro la complessa matematica statistica di questo primitivo “mostro cibernetico” rappresentato del sistema antiaereo, si nascondeva un nucleo perturbante umanità che è stato necessario approfondire: l'*unheimlich* di un meccanismo che, pur privo di volto, sembrava ragionare e prevedere, come mostrano le testimonianze dirette dei superiori di Wiener nelle gerarchie nell'apparato scientifico-bellico. In questa zona d'ambiguità prende forma una riflessione intorno all'*antropomorfismo* delle macchine [Roselli et al., 2025], ossia l'automatica proiezione di qualità umane in entità non umane o artificiali.

Il punto da indagare è stato quindi allora come le caratteristiche tecniche del predittore – l'uso di dati, la logica del *feedback*, la chiusura del ciclo di retroazione – abbiano contribuito a definire una visione dell'uomo, le condizioni culturali di questa visione ipersemplificata dell'umano, resa plausibile solo dal contesto bellico. Come ha mostrato Galison [Galison, 1993], tale visione ha lasciato un preciso *significato culturale* che ha lasciato, e che pone un quesito analizzato in questo lavoro sulle problematichità delle tecnologie digitali contemporanee – l'Intelligenza Artificiale connessionista protagonista di questa “estate” del digitale – che si fondano sulla stessa natura riduzionistica e probabilistica dei primigeni concetti cibernetici,

dalle opacità intrinseche dei sistemi di *machine learning* e *deep learning*, all'astrazione della dimensione della corporeità, alle problematicità della logica delle *black box* e di generalizzazione statistica, istanze sempre più decisive in ogni aspetto della vita quotidiana di ciascuno: dalla l'accesso al welfare, al credito, alla selezione per un impiego. Ma gli algoritmi incidono anche nella sfera della quotidianità e persino in quella affettiva: dalla scelta di un ristorante a quale film guardare, dalla musica da ascoltare fino alla selezione di un potenziale partner [Crawford, 2021; Liao, 2020; O'Neil, 2016; Pasquale, 2015].

Ma la cibernetica, all'interno di una dinamica al tempo stesso culturale e contro culturale - che ritroveremo in forme diverse e che sarà indagata a più livelli nel corso di questo lavoro - oscillò tra, da un lato, la paura distopica di esseri umani ridotti, come semplici dati di un gigantesco organismo informazionale, a schede perforate IBM - emblematicamente richiamate dallo slogan "*Don't Spindle, Fold or Mutilate Me*" esibito dagli studenti delle contestazioni nelle università californiane - e, dall'altro, una visione di liberazione cognitiva, in cui gli artefatti cibernetici e gli scritti di Wiener venivano accolti dai *New Communalists*, da artisti, ricercatori e tecnoutopisti delle comuni californiane come strumenti di emancipazione della mente e di amplificazione delle sue capacità [Turner, 2006].

È stato quindi necessario prendere in esame, dal punto di vista tecnico, teorico e della sua genesi storica, il concetto di *informazione*, inseparabile da quello di cibernetica e fondamentale per comprendere la nascita del pensiero computazionale moderno. Elaborato da un altro dei protagonisti di questa storia, Claude Shannon [Shannon, 1948], il concetto di informazione fu definito in modo matematicamente rigoroso e legato al principio di entropia: una misura quantitativa dell'incertezza, che avrebbe poi assunto, come vedremo, anche una rilevanza etica nella filosofia dell'informazione contemporanea [Floridi, 2009].

Shannon formulò un modello di comunicazione indifferente al contenuto semantico, potenzialmente applicabile al rapporto tra qualunque sistema di informazione – artificiale o umano – in grado di trasmettere e ricevere segnali. Come per Wiener, anche la sua riflessione nacque nel contesto bellico, durante il lavoro di crittografia per l'esercito statunitense. Inoltre nella celebre tesi di master del 1937 [Shannon, 1939, Gardner, 1985] – realizzò per la prima volta un'identificazione funzionale fra una componente biologica e una artificiale: i relè elettrici capaci di eseguire le funzioni logiche booleane, equivalenti alle operazioni elementari dei neuroni. Questi dispositivi – una sorta di *micromondo* digitale in un gigante *analogico*, l'*Analizzatore Differenziale* di Vannevar Bush - rivelavano già la possibilità e la forza della

metafora cibernetica: la concezione dell'uomo e della macchina come sistemi isofunzionali di elaborazione e controllo dell'informazione.

In tale equivalenza, che da semplice analogia diventa oggetto di indagine, prende forma una delle domande centrali di questo lavoro: quali sono i limiti – tecnici, implementativi e teorico-filosofici – dell'analogia funzionale tra “cervello” o “mente” e macchina come sistemi di computazione dell'informazione?

Analizzando, dal punto di vista storico, tecnico e teorico, le analogie e le differenze del modello “Wiener-Shannon” [Wiener, 1956] di informazione – contrariamente a quanto sostiene, ad esempio, Hayles [Hayles, 1999], che ne sottolinea la sostanziale identità – emerge come la maggiore attenzione data da Wiener alla dimensione semantica dell'informazione abbia prefigurato modelli somatotropici e neuromorfici del cervello [Arbib, 1972], destinati a rivelarsi decisivi nello sviluppo della moderna IA “neurale”.

Inoltre, quasi per inciso, è solo concependolo come sistema di informazione che diventa comprensibile anche la pretesa “controculturale” – emersa negli anni Sessanta e Settanta – di considerare il cervello un sistema di *augmenting* delle facoltà cognitive e sensoriali dell'essere umano.

Comprendere “*tutto il sistema*”, come recitava una delle sezioni di una pubblicazione chiave della controcultura americana degli anni '60, lo *Whole Earth Catalog* [Brand, ed. 1968], significa comprendere le modalità di interazione via via più profonde e personali tra uomo, macchine e ambiente, che spiegano come alcuni abbiano sognato delle *Machines of Loving Grace*, prefigurazione tra il poetico e l'utopico dei tentativi di creare un'*ecologia dell'informazione* eticamente sostenibile della contemporaneità.

Macchine di Turing di carne?

Da questa prospettiva diverrà imprescindibile, nel secondo capitolo, allora porre l'analisi ad un livello d'astrazione più elevato e prendere in esame i limiti delle capacità computazionali della mente e del calcolatore, e del loro rapporto [Marconi, 2025], facendone oggetto di interrogazione critica.

Oggetto di questa indagine non poteva che essere Alan Turing, che nel 1936, con *On Computable Numbers, with an Application to the Entscheidungsproblem* [Turing, 1936], formulò il modello teorico destinato a diventare la base di ogni calcolatore digitale: la macchina universale. Verrà mostrato come una nozione apparentemente “primitiva” delle tecnologie digitali - quella di algoritmo, di cui Turing fornisce la prima definizione rigorosa nella cosiddetta

tesi di Church-Turing - presupponga la presenza di un soggetto umano che la interroga, ne fornisce le risorse materiali e ne interpreta le risposte [Dodig-Crnkovic, 2011].

Nel lavoro di Turing - dal testo fondamentale del 1936 alla tesi di dottorato *Systems of Logic Based on Ordinals* [Turing, 1939], fino a *Intelligent Machinery* [Turing, 1948] - emerge una tensione costante tra la manipolazione sintattica dei simboli, propria delle macchine universali, e l'ambizione di "costruire un cervello", ossia di dotare le macchine delle facoltà di intelligenza proprie del cervello umano, nei termini di memoria e apprendimento.

In questa prospettiva, Turing appare come un proto-conneccionista, anticipando l'idea che l'intelligenza sia un processo distribuito e auto-organizzativo, più vicino al funzionamento neurale che alla pura esecuzione algoritmica [Cfr. Copeland, Proudfoot 1996;1999; 2017; Copeland 2004cc, pp. 403-409; Dewhurst, 2019; Sprevak, 2017; Numerico, 2005]

A tal fine saranno analizzate le macchine non organizzate di Turing - le sue *child-machines* - e l'*Imitation Game* (il celebre "*test di Turing*"), intesi come tentativi di riprodurre funzionalmente alcune capacità del cervello umano in sostrati materiali diversi [Cfr. Copeland, 2000; Moor, 1976; Moor, 2001].

L'analisi di questi punti permette di comprendere come le tesi di Turing e Church sulla computabilità non rappresentino soltanto una capacità, ma anche un limite: "una fondamentale scoperta circa le limitazioni del potere matematizzante dell'Homo sapiens" [Post, 1936].

Qui si apre uno spiraglio interpretativo che getta luce sui paradigmi concorrenti dell'Intelligenza Artificiale e sulle prospettive di apprendimento delle macchine: da un lato, il programma "classico" della cosiddetta GOFAI (*Good Old-Fashioned Artificial Intelligence*), fondato sulla manipolazione simbolica e sulla rappresentazione esplicita di regole (e base delle scienze cognitive, che concepiscono la mente come elaboratore di informazioni) [Haugeland, 1985; Newell & Simon, 1976]; dall'altro, il paradigma conneccionista, che considera l'intelligenza come il risultato di processi emergenti e adattivi.

Da qui discendono le domande che orientano la riflessione che seguirà: fino a che punto è possibile costruire una macchina capace di apprendere? Che cosa rimane irriducibile nel funzionamento del cervello umano rispetto ai modelli computazionali che tentano di riprodurlo? E quale immagine dell'intelligenza - e del suo sostrato - emerge da questa tensione tra calcolo e apprendimento, tra formalizzazione e plasticità neurale? È possibile tradurre tale dinamica in hardware, mantenendo al tempo stesso il rigore logico-matematico che fonda la computazione?

In hardware: direttive storiche e sviluppi tecnologici

Gli anni Quaranta furono un decennio in cui emersero, con forza inedita, domande fondamentali sul modo in cui l'essere umano dovesse interagire con macchine di controllo sempre più complesse. Questo è il periodo, che sarà preso in esame - in linea con le direttive della storiografia della computazione [Haigh, Ceruzzi, 2021; Haigh et al., 2019; Ceruzzi, 2012] - in cui è avvenuta la progressiva trasformazione del *computer* - l'essere umano, più spesso una donna, impegnata in calcoli a matita e penna - nel *computor* [Mauchley, 1942]: l'artefatto, spesso di proporzioni *colossali*, capace di eseguire automaticamente, e con una rapidità inimmaginabile, operazioni prima riservate all'intelligenza umana.

Posto il quadro prospettico della metafora cibernetica e indagati i concetti di informazione e computabilità, diventa necessario scendere nel piano concreto dell'implementazione delle macchine che elaborano l'informazione. L'obiettivo sarà quindi comprendere come le idee e le realizzazioni di Shannon, Wiener e Turing abbiano trovato corpo in dispositivi materiali che, nati per finalità belliche, hanno trasformato in profondità la relazione tra pensiero, macchina e conoscenza.

Proprio in questi anni saranno individuati dei vettori di sviluppo capaci di dare un quadro interpretativo alle tecnologie digitali, del loro *design* e al loro rapporto con l'essere umano, indagati e sviluppati in quattro direzioni principali [Ceruzzi, 2012]: *Sviluppo del paradigma digitale*; *convergenza tecnologica*: la progressiva integrazione di differenti traiettorie di sviluppo - elettronica, telecomunicazioni, calcolo - rese compatibili dal linguaggio comune del digitale; *Elettronica a stato solido*: Il passaggio dai relè ai transistor e ai circuiti integrati - come analoghi funzionali delle componenti del cervello umano - caratterizzato da una miniaturizzazione e da un aumento di efficienza progressivi, apparentemente guidati dalla cosiddetta "legge di Moore". Tale prospettiva, tuttavia, come vedremo, mette in luce le contraddizioni e smentisce il determinismo tecnologico che ha a lungo guidato la narrazione dello sviluppo informatico; *principio del programma memorizzato*: L'*embodiment* delle idee turinghiane sulla computazione e sull'intelligenza delle macchine, che consente all'elaboratore di trattare istruzioni e dati nello stesso spazio di memoria, aprendo la strada alla flessibilità e alla generalità del calcolo automatico, rendendo dati e istruzioni equivalenti; *interfaccia uomo-macchina*: l'insieme delle modalità di relazione e cooperazione tra essere umano e tecnologie digitali, che interrogano il confine tra sostituzione meccanica e estensione cognitiva dell'uomo.

Nel passaggio dal paradigma analogico a quello digitale si compie una svolta decisiva. L'analogico - dominio della continuità, dei segnali variabili e dei calcolatori differenziali di Vannevar Bush - lascia spazio a un nuovo modello fondato sulla discrezione, sulla codifica binaria e sull'idea che ogni fenomeno possa essere rappresentato mediante due soli stati: 0 e 1. I due simboli impressi sul nastro della macchina di Turing, i due valori booleani del calcolo logico (vero e falso), o i due stati elettrici di un circuito (aperto e chiuso), divengono il linguaggio universale di un mondo riscritto in termini digitali. La caccia all'equazione differenziale, necessitava, dato il contesto di crescente richiesta computazionale per fini balistici, un nuovo metodo.

In questo quadro, via via, la metafora cibernetica si incarna in dispositivi tecnologici concreti: le valvole termoioniche, i tubi catodici, e le prime architetture elettroniche capaci di impiegare "velocità elettroniche" nell'implementare le funzioni logiche prima riservate al cervello umano.

In queste macchine - costruite per rispondere a esigenze belliche di calcolo e predizione - si delinea, in *nuce*, una relazione simbiotica tra operatore umano e apparato computazionale, un sistema integrato di percezione e risposta che traduce in forma tecnica l'intuizione cibernetica.

A Bletchley Park, la logica formale e la meccanica elettrica si incontrano nei sistemi di decifrazione *Heath Robinson* e *Colossus*, costruiti per violare i cifrari tedeschi. In essi, la matematica della probabilità e la logica booleana si traducono in operazioni elettroniche, inaugurando la computazione automatica.

L'interazione fra l'intuizione umana degli analisti e la velocità dei circuiti mostra per la prima volta come la collaborazione tra mente e macchina possa superare i limiti di entrambe. Negli Stati Uniti, le stesse esigenze di calcolo e previsione producono le prime macchine completamente elettroniche: l'ENIAC (1945) e, poco dopo, l'EDVAC, che introduce il principio del programma memorizzato, l'*incarnazione* - inquadrata in un quadro biologista-cibernetico ispirato da McCulloch e Pitts - delle idee di Turing sui calcolatori

Quest'ultimo segna un punto di svolta: il suo schema logico, redatto sotto la guida di un altro grande protagonista di questa storia, John von Neumann, sulla base dei lavori di Mauchly ed Eckert, racchiude la struttura di ogni calcolatore successivo, in cui dati e istruzioni condividono la stessa memoria. L'EDVAC diventa così il paradigma incarnato della metafora cibernetica, un modello in cui la mente e la macchina si riflettono come sistemi equivalenti di controllo e comunicazione.

Il contributo di Turing, in questi stessi anni, si concentra sui tentativi di dare corpo alle proprie idee sull'intelligenza delle macchine. Il progetto ACE (*Automatic Computing Engine*), elaborato per il *National Physical Laboratory*, rappresenta uno dei primi disegni completi di un calcolatore basato su principi logico-matematici formalizzati: un passo concreto nella direzione della macchina universale. Parallelamente, la collaborazione con il gruppo di Manchester porta alla realizzazione della *Manchester Baby*, il primo calcolatore elettronico a programma memorizzato effettivamente funzionante.

In queste realizzazioni si riflettono gli sforzi di Turing di esplorare le possibilità e i limiti di una mente meccanica, capace di rappresentare e manipolare informazioni secondo principi logici e di apprendimento.

Questi sviluppi, nati nel contesto bellico e proseguiti nell'immediato dopoguerra, segnano un arco di maturazione e criticità che condurrà, negli anni successivi, al superamento del paradigma incarnato dall'EDVAC.

Da qui prende avvio un nuovo capitolo: la progressiva miniaturizzazione delle macchine, la loro diffusione fuori dai laboratori militari e accademici, e il trasferimento del centro di gravità della ricerca verso la California, dove la cibernetica e la logica dei calcolatori troveranno, come vedremo, nuove vie di sviluppo nella computazione personalizzata

La ricostruzione di questa dinamica si è avvalsa delle testimonianze e delle ricostruzioni storiche di alcuni protagonisti diretti dello sviluppo delle tecnologie computazionali [Goldstine, 1972; Metropolis et al., 1980; Randell, 1975], insieme a relazioni e dossier tecnici - spesso rimasti secretati per decenni - come il *General Report on Tunny* redatto da Jack Good, Donald Michie e Geoffrey Timms [Good, Michie, Timms, 1945]. A queste fonti si affiancano analisi e relazioni tecniche sulle tecnologie impiegate, relative ai dispositivi materiali dei primi calcolatori - come le valvole termoioniche e i tubi catodici - e ai manuali operativi elaborati negli stessi anni, tra cui il *Programmer's Handbook* redatto da Alan Turing. L'intera ricostruzione si colloca nel quadro interpretativo storiografico e storico-scientifico più recente, volto a chiarire l'intreccio tra sviluppo tecnologico e pratiche di ricerca che hanno accompagnato la nascita del calcolo automatico.

L'uso congiunto di queste fonti e dei contributi successivi ha permesso, nel terzo e nel quarto capitolo di questo lavoro, non solo di restituire, con precisione storica, la complessità dei processi che condussero alla nascita della computazione moderna, ma anche di chiarire da un punto di vista critico-filosofico - alla luce della riflessione contemporanea - nozioni

fondamentali come “*informazione*” e “*dato*”, rendendone più comprensibile la genesi storica e la centralità nella cultura della modernità.

Un mondo chiuso, una rivoluzione personale

Una volta indagato lo sviluppo storico dei primi calcolatori, l'evoluzione del loro design e delle loro potenzialità tecniche nel rapporto costante con l'essere umano che li utilizzava, la domanda si declina su un nuovo piano: che cosa accade quando il paradigma che li ha generati si diffonde, oltre i laboratori e gli arsenali, nel tessuto stesso della società?

La diffusione del paradigma EDVAC - ispirato a una ben determinata visione dell'elaborazione dell'informazione umana, concepita come processo seriale, discreto e quantificabile in bit - segnò l'avvio di una trasformazione profonda. Ciò che, nei laboratori di Princeton e della Moore School, era stato teorizzato come architettura logico-matematica, sebbene originariamente legata a logiche belliche, divenne progressivamente un modello operativo di assetto e di governo: non più soltanto un principio tecnico, ma una forma di organizzazione della società nel suo complesso interpretata come un organismo.

Nel quinto capitolo ci muoveremo allora lungo queste domande, per mostrare come la diffusione del paradigma cibernetico, lungi dall'essere un processo tecnico o scientifico neutro, abbia risposto a precise esigenze socio-politiche e culturali. Come ha evidenziato Paul N. Edwards, tanto l'ingegneria centrata sul calcolatore quanto la politica del “mondo chiuso” della Guerra Fredda ruotavano attorno a un unico problema: integrare uomo e macchina in sistemi di previsione, difesa e governo capaci di reagire su scala globale e nei tempi estremamente compressi della deterrenza nucleare [Edwards, 1996]. A questa tensione si accompagnò una duplice dinamica: da un lato, la liberazione dalla paura del pericolo atomico, che alimentava la fiducia in una razionalità tecnico-sistemica capace di garantire la sicurezza collettiva; dall'altro, la liberazione dal gravoso lavoro matematico, resa possibile da una crescente automatizzazione del calcolo e da forme di comunicazione con la macchina sempre più vicine al linguaggio umano.

Nato come strumento di calcolo, il computer si è trasformato progressivamente in strumento di coordinamento e di regolazione, incarnando l'idea di una società omeostatica, autoregolata, in cui informazione e decisione circolavano come flussi vitali all'interno di un corpo collettivo. La metafora biologista della cibernetica, che interpretava il corpo sociale come un organismo attraversato da circuiti di retroazione, stabilità e adattamento, trovò nel periodo post-bellico, in questa visione, la sua piena legittimazione.

Ma in che modo questa visione dell'automazione come destino collettivo, sostenuta dalla rappresentazione del corpo sociale come organismo integrato, si è diffusa con lo sviluppo del calcolatore come strumento commerciale e scientifico sempre più presente nella vita civile? E come, attraverso i grandi sistemi computazionali della Guerra Fredda - dal sistema SAGE, simbolo della difesa automatizzata e del "mondo chiuso", al successo dell'IBM, che contribuì a diffondere non solo la tecnologia ma la metafora stessa che la sosteneva - il computer divenne l'architrave di una società concepita come organismo informazionale?

Indagheremo dunque la natura "sociologica" della metafora cibernetica: dall'impiego dei concetti chiave come omeostasi e feedback, interpretati, nella lettura di Talcott Parsons, come meccanismi di stabilizzazione e conservazione dello status quo, fino alle idee già incontrate in Wiener, che ne rivendicavano invece un uso emancipativo, volto a una nuova alleanza comunicativa tra esseri umani e macchine, chiave di volta per comprendere caratteristiche chiave della nostra società della *Quarta Rivoluzione* [Floridi, 2014; Heims, 1991, Kline, 2015; Bynum, 2008; Wiener, 1950; 1948].

La via della personalizzazione e dell'elaborazione distribuita è stata indagata seguendo la direttrice dello sviluppo dell'elettronica a stato solido. La via che porta alla Silicon Valley è stata percorsa su una strada di elettroni. L'analisi storica della materializzazione elettronica - dal transistor al microchip - condotta attraverso il lavoro dei loro inventori, ha permesso di mettere in luce un fenomeno decisivo: l'inversione del determinismo tecnologico, dotando, al contempo, le riflessioni controculturali di concretezza tecnico-scientifica.

Parallelamente, verranno evidenziati i mutamenti di paradigma nella struttura fisica e concettuale del calcolatore: dai *mainframe*, simboli della centralizzazione informatica, al *time-sharing*, figlio di una integrazione simbiotica tra uomo e macchina, fino al *personal computer*, in cui il calcolatore si trasforma progressivamente in strumento di comunicazione.

A questo proposito, l'analisi del capitolo non poteva che prendere avvio da una ricostruzione delle testimonianze dei protagonisti, come Stanislaw Ulam e Nicholas Metropolis, che per primi hanno aperto la strada a un nuovo universo: quello della simulazione, scaturito dalle applicazioni del metodo Monte Carlo necessarie ai calcoli per la bomba all'idrogeno. Un processo che ha trasformato il computer da strumento di calcolo a elemento costitutivo di una realtà computerizzata.

Questa analisi si rivela necessaria non solo per mettere in luce le *trading zones* e il *pidgin* - un linguaggio creolo e ibrido - su cui si è strutturata e diffusa la retorica della computazione [Galison, 1996], ma anche per comprendere una delle categorie chiave della contemporaneità,

quella dei *big data*, e per metterne in luce le implicazioni epistemologiche e persino ontologiche: il passaggio da una rappresentazione del reale a una sua generazione digitale. In questa dinamica emerge via via il ruolo del programmatore come designer di un nuovo mondo, sempre più ricco e sempre più abitabile.

In parallelo agli sviluppi della computazione, saranno esaminati, inoltre, dal punto di vista dell'interazione uomo-macchina, la nascita e gli sviluppi dei linguaggi di programmazione, intesi come forme di ausilio e di delega di prerogative umane ai calcolatori, con figure centrali come Grace Hopper, in un progressivo mutare di livello della gestione di un concetto centrale: quello della *complessità*, e del passaggio di forme di comunicazione sempre più vicine, da quello fatto di 0 e 1 delle macchine, a quello naturale dell'essere umano.

La cibernetica di fronte al suo tribunale

L'analisi, svolta nel capitolo quarto, poneva una domanda sull'appropriatezza dell'impiego della metafora cibernetica rispetto alle caratteristiche di design dell'architettura di von Neumann. L'indagine, dopo aver analizzato il livello "macro" della diffusione - e della retorica - di questo paradigma, scende per rispondervi, nel sesto capitolo, al livello "micro", nella struttura stessa degli elementi che costituiscono l'equivalenza alla base della metafora cibernetica.

Protagonista, in modo solo apparentemente paradossale, sarà lo stesso John von Neumann, del quale verranno analizzate innanzitutto le perplessità nei confronti dell'apparato logico-matematico - quello di Turing e di McCulloch-Pitts - che ne costituiva il fondamento, espresse in una lunga lettera a Norbert Wiener, da cui prenderà forma un nuovo programma di ricerca. Il problema centrale è quello della *complessità*: la cibernetica, posta di fronte al proprio tribunale, rivela come l'integrazione nel cuore delle sue componenti costitutive dell'informazione shannoniana - l'elemento che la parte biologica e quella artificiale processano - richieda una riforma della logica che la sostiene.

Questa trasformazione, analizzata attraverso la teoria degli automi proposta dal matematico ungherese, ci consentirà di mettere in luce il passaggio da una logica combinatoria a una logica probabilistica, e, come in Wiener, orientata in direzione *neuromorfica*, nella gestione della complessità dell'"automa-cervello".

Come emergerà da *The Computer and the Brain*, il pensiero di von Neumann si orienterà verso una concezione del calcolo in cui il *parallelismo*, il carattere solo parzialmente digitale, la

natura stocastico-probabilistica e addirittura il carattere *storico* del linguaggio logico-matematico della computazione diventano tratti essenziali.

L'indagine, in questo senso, si propone di mostrare come le riflessioni di von Neumann sui limiti dell'architettura a programma memorizzato - i cosiddetti "colli di bottiglia" - conducano non solo a una critica interna alla metafora cibernetica, ma anche a individuare alcune proprietà che costituiranno la base dei futuri sviluppi della computazione e dell'intelligenza artificiale, in un parallelo con le riflessioni già incontrate in Turing sulle *macchine non organizzate*.

Neuroni artificiali, verso l'AI contemporanea

Una volta individuato il retroterra concettuale nel cuore stesso della cibernetica, il passo successivo è comprendere come, a partire da quel nucleo teorico, si siano sviluppati tentativi di computazione alternativa al modello seriale di von Neumann: esperimenti di una razionalità del calcolo fondata sul parallelismo, sull'auto-organizzazione e sull'apprendimento distribuito.

L'analisi storica condotta mostrerà come l'"antenato" dell'attuale Intelligenza Artificiale connessionista debba essere riconosciuto nel *Perceptron* di Frank Rosenblatt [1958], espressione di una linea di ricerca che, dall'idea hebbiana di apprendimento sinaptico [Hebb, 1949] alla visione probabilistica di von Neumann, segna un progressivo allontanamento dalla logica deterministica e simbolica del calcolatore. Da quel modello - fragile e visionario insieme - prende forma una genealogia di architetture e di ipotesi - sebbene sepolti sotto il "gelo" di un rigido inverno dell'IA - che saranno ricostruite nel lavoro che segue dal punto di vista storico, matematico e filosofico: dal *Pandemonium* di Selfridge (1959) all'Adaline di Widrow e Hoff (1960), fino ai modelli auto-organizzativi di Kohonen, Grossberg, Amari e Ivakhnenko [Cordeschi, 1998; Medler, 1998; Buckner, Garson, 2019].

In questi sviluppi, il neurone e la sinapsi divengono i nuovi criteri organizzativi del calcolo, e la rete stessa il principio strutturale e funzionale del sistema: un ordine distribuito e plastico che sostituisce la rigida gerarchia del design di von Neumann, istanza che non ha solo conseguenze tecnologiche, ma anche profonde implicazioni culturali e cognitive.

Su queste basi si innesta la rinascita degli anni Ottanta, con l'introduzione della *backpropagation* da parte di Rumelhart, Hinton e Williams [1986] e la pubblicazione di *Parallel Distributed Processing* di McClelland, Rumelhart e Hinton [1986], manifesto teorico di un nuovo paradigma cognitivo e computazionale. Le reti multilayer - capaci di apprendere e generalizzare attraverso la modulazione dei pesi e la retropropagazione dell'errore - segnano, come sarà

preso in esame, un cambio di prospettiva: dal simbolo alla connessione, dall'algoritmo alla plasticità, dall'ordine imposto alla forma emergente. Parallelamente, l'hardware cerca di incarnare questa nuova visione del calcolo nelle *Connection Machines* di Danny Hillis [1985], supercomputer a parallelismo massivo che rappresentano il tentativo più radicale di tradurre in design materiale il principio della distribuzione cognitiva.

Ed è da questo orizzonte che prende avvio la rivoluzione del *deep learning*, di cui prenderemo in esame una panoramica delle conseguenze negli sviluppi dell'IA attuale, come erede diretto della genealogia qui tratteggiata, e guidata dai cosiddetti "tre padrini" dell'intelligenza artificiale contemporanea - Hinton, LeCun e Bengio - che hanno portato le intuizioni della cibernetica verso reti neurali di capacità e profondità sempre maggiori.

Questa traiettoria sembra condurre a compimento le intuizioni di Turing, von Neumann e Wiener: l'idea di una mente - o, più propriamente, della costruzione di un "cervello" artificiale - capace di apprendere, ricordare e riconoscere.

Ma ciò che oggi chiamiamo *intelligenza artificiale* è davvero intelligenza, o piuttosto questa idea rappresenta l'ultimo riflesso dell'antica ambizione dell'uomo di comprendere sé stesso costruendo una macchina a sua immagine?

Dall'intelligenza all'azione

Il lavoro non può che avviarsi alla conclusione a partire da questa domanda. Al di là del tenore, sopra menzionato, del discorso pubblico-mediatico e della pervasività delle tecnologie digitali e dell'intelligenza artificiale contemporanea, risulta indispensabile metterne in luce rischi e criticità. Accanto alle indubbie potenzialità, tali tecnologie presentano infatti problematicità che è ineludibile tenere presenti.

I modelli linguistici di grandi dimensioni, che costituiscono il cuore dell'attuale fase dello sviluppo dell'IA, si fondano su una logica puramente statistica di predizione: non comprendono, ma correlano; non interpretano, ma riproducono schemi probabilistici di associazione. Questa struttura li rende capaci di generare testi e immagini coerenti soltanto in apparenza, producendo un linguaggio che simula il senso senza possederlo.

Ne derivano rischi profondi, che investono al tempo stesso il piano tecnico, cognitivo e sociale: una radicale opacità epistemica, dovuta alla scala e alla complessità dei modelli, che ne rende di fatto inaccessibili i processi interni; la riproduzione sistemica dei bias presenti nei dati di addestramento, con la conseguente amplificazione di disuguaglianze e stereotipi; l'impatto ambientale crescente, legato ai costi energetici e materiali dell'addestramento; e infine

l'illusione di comprensione, la tendenza a scambiare la coerenza statistica per intelligenza, la fluidità linguistica per pensiero.

A queste si aggiungono ulteriori difficoltà strutturali: la dipendenza massiccia dai dati, che rende tali sistemi efficaci solo laddove dispongano di quantità enormi di esempi; la fragilità inferenziale, che li limita nei compiti aperti o non rigidamente circoscritti, dove è richiesta la capacità di integrare conoscenze implicite e contesti nuovi; e la mancanza di continuità cognitiva con la conoscenza umana, poiché questi modelli apprendono in modo isolato, ricostruendo statisticamente regolarità che per l'essere umano derivano invece da concettualizzazione e ragionamento. [Bender et al., 2021; Marcus, 2018].

A partire da ciò e dai fallimenti dell'IA di essere cognitivamente riproduttiva delle capacità di intelligenza umana (o sub-umana), non si può che riconoscere in essa una "riserva ingegnerizzata della capacità d'agire" [Floridi, 2022]. L'attenzione quindi deve spostarsi sulla dimensione etica.

Pragmatica dell'informazione diffusa: l'eticista designer

Questo *shift* sulla dimensione etica chiude le analisi di questa indagine. Come è possibile, alla luce delle mutate caratteristiche della fenomenologia etica tradizionale causate dalle nuove tecnologie [Van den Hoven, 2017] dare corpo a valori morali condivisi, quelli *dell'innovazione sostenibile* [Umbrello, 2023] e dell'*Artificial Intelligence for Social Good* (AI4SG) [Floridi, 2022], - orientati alla prevenzione, mitigazione o risoluzione di problemi sociali e ambientali - possono trasformarsi in strumenti pratici di progettazione e valutazione?

La soluzione che qui si proporrà prende forma nell'individuazione di una figura professionale emergente, della quale saranno individuate le caratteristiche, quella dell'*eticista-designer* [van Wynsberghe, Robbins, 2013], chiamata a tradurre nella pratica i valori morali condivisi tra la pluralità di *stakeholder*- umani a artificiali [Floridi, Sanders, 2004] - che mutano il quadro della responsabilità individuale erede della tradizione moderna [Flusser, 2003; Verbeek, 2006], popolano gli ambienti dove le tecnologie digitali vengono ideate, progettate implementate.

La ricostruzione complessiva del lavoro che seguirà, dalle origini dell'analogia funzionale tra dimensione artificiale e biologica, all'analisi dei limiti della computazione allo sviluppo delle prime macchine computazionali sino all'IA contemporanea - nel quadro delle dinamiche sociotecniche descritte lungo tutto il percorso - ha assunto come punto di vista critico-prospettico l'idea di una comune *struttura informazionale* che accomuna ogni sistema.

A tal fine si è ricostruita la “metafora cibernetica” e il suo *design* nella sua traiettoria storica. Nel quadro della contemporaneità, una relazione uomo-macchina che sfugga dalla dicotomia utopia-distopia e sappia essere una proficua riserva d’azione per l’essere umano, capace di preservarne le caratteristiche, il τί ἐστὶ, deve tradurre la storia e le conseguenze concettuali del sorgere delle tecnologie digitali in una *pragmatica* dell’informazione che sappia considerare quest’ultima “come livello d’astrazione comune per la considerazione di ogni ente, il quale è a un livello ontologico minimale non contraddittorio e denotabile” [Floridi, 2009] in quadro dove ogni entità come “specie non umane, superorganismi, la Terra, i social robot” [Friedman, Hendry, 2019, p.27]⁴ hanno una comune matrice nell’informazione e devono essere messi in comunicazione.

A questo fine, il *fil rouge* attorno al quale il lavoro che segue sarà intrecciato lavoro sarà quello individuare i capisaldi storici e concettuali, i limiti, le potenzialità e le false pretese con cui questa dinamica si è concretizzata, in una dinamica che, in conclusione, trova una genealogia diretta tra le considerazioni della prima cibernetica wieneriana agli ultimi sviluppi della *Information Ethics* [Cfr. Bynum, 2009], a conclusione di quel cerchio che ha visto la scienza dell’uniformità dell’uomo e della macchina sempre al crocevia tra disumanizzazione ed emancipazione,

Una parola chiave della contemporaneità è quella di algoritmo. Gli algoritmi strutturano oggi le tecnologie della comunicazione, ma, oltre alla loro funzione tecnica, vanno considerati, nella loro originaria simbiosi con l’essere umano, come vere e proprie “macchine culturali”: “complessi aggregati di astrazioni, processi e persone” che affondano “le loro radici non solo nella logica matematica, ma anche nelle tradizioni filosofiche della cibernetica, della coscienza e della magia del linguaggio simbolico” [Finn, 2017, p. viii, trad. it.]. Le modalità della loro implementazione e della loro analisi devono dunque mantenere una dimensione critica, poiché “comprendere come conoscerla richiede i metodi critici delle discipline umanistiche” [Finn, 2017, p. viii, trad. it.].

Trovare grazie a questi metodi critici il modo di disinnescare le *bombe* (la tecnologia che più di tutte ha fatto *detonare* l’universo digitale) e trasformarle in valori capaci di lasciar intravedere una sagoma umana dietro gli schermi degli apparecchi digitali è ciò che questo lavoro si propone.

⁴ "The term "entity" is used here to account for nonhuman stakeholders such as an organization, nonhuman species, or sacred mountaintop"

A tal fine, in chiusura di questo lavoro, prima di trarre le conclusioni, sarà collocato il framework operativo sviluppato presso Exprivia S.p.A.; si ringraziano il dott. Vitantonio Fascilla, il dott. Nicola Tino, la dott.ssa Tiziana Giove e il dott. Domenico Favuzzi per avermi concesso la possibilità, rara e privilegiata, di sperimentare *in vivo* e di rendere concrete, in un contesto d'eccezione, le riflessioni qui elaborate, là dove le tecnologie digitali oggetto dell'analisi vengono effettivamente progettate e trovano concreta attuazione. Il framework, progettato graficamente per essere immediatamente fruibile da ogni operatore coinvolto nell'implementazione delle tecnologie digitali, costituisce una sintesi applicativa dei principi teorici, storici ed etico-critici ricostruiti nel presente studio.

In esso trovano posto tabelle di principi guida, schemi di riferimento per la progettazione responsabile, linee di buone pratiche (*best practices*), criteri di fairness e elenchi di rischi, criticità e malpractice da evitare nelle fasi di sviluppo degli algoritmi di Intelligenza Artificiale e nel design delle interfacce uomo-macchina.

Strutturato come strumento di consultazione e di riflessione operativa, il framework intende offrire un modello pratico per l'implementazione etica e sostenibile delle tecnologie digitali. L'obiettivo è tradurre i valori etici in criteri verificabili e in procedure di design, rendendo così l'etica una dimensione costitutiva - e non accessoria - dell'innovazione tecnologica. In tal modo, il framework si pone in continuità con il quadro normativo internazionale e nazionale, ma anche con i valori propri dell'impresa e del singolo operatore tecnologico, promuovendo una cultura della responsabilità progettuale come parte integrante del processo innovativo.

L'itinerario che nelle pagine seguenti si è tentato di ricostruire - dall'indistinzione tra nervi e acciaio, dalle valvole incandescenti dei primi calcolatori al silicio dei transistor, dagli elettroni alle sinapsi, dalle macchine logiche di Turing ai sistemi che oggi apprendono, ricordano e generano contenuti - è un percorso tracciato attraverso fonti storiche, rapporti tecnici, testimonianze dirette e analisi concettuali, volto a restituire, oltre la superficie delle architetture e dei codici, la trama di significati che lega la storia delle macchine alla storia dell'uomo. Perché, al di là dei modelli e delle simulazioni, il fine ultimo di questa indagine è proprio questo: far riemergere dal groviglio di cavi, spinotti, transistor e microchip e materializzare dall'insieme dei bit, l'immagine dell'essere umano - non come residuo del passato, ma come principio generativo del suo stesso futuro digitale.

Cap. 1: *Understanding Whole System*: Intorno alle radici scientifico-culturali del rapporto uomo-macchina. Dall'archeologia cibernetica all'idea di personalizzazione

1. Scarti della macchina e poesie d'amore elettronico

a. "Sono uno studente: non piegare, infilzare o mutilare"

Nell'estate del 1967 – la *Summer of Love* - nel luogo che rappresentava il cuore pulsante dell'estate simbolo della controcultura, per le strade del distretto di Height Ashbury a San Francisco, in California, Richard Brautigan (1935-1984), un giovane poeta "*long-haired*", distribuiva a mano ai passanti un grande foglio mimeografato, formato "lenzuolo", su cui era scritta una poesia, intitolata, *All Watched by Machines of Loving Grace*⁵ [fig.1], che recitava:

I like to think (and
the sooner the better!)
of a cybernetic meadow
where mammals and computers
live together in mutually
programming harmony
like pure water
touching clear sky.
I like to think
(right now, please!)

5 "Mi piace pensare (e prima è meglio è!) / a un prato cibernetico / dove mammiferi e computer / vivono insieme in un'armonia programmata reciprocamente, / come acqua pura che tocca un cielo limpido. / Mi piace pensare (proprio adesso, per favore!) / a una foresta cibernetica / piena di pini ed elettronica / dove i cervi passeggiano tranquillamente / accanto ai computer, / come fossero fiori con petali rotanti. / Mi piace pensare (deve essere così!) / a un'ecologia cibernetica / dove siamo liberi dalle nostre fatiche / e riconnessi alla natura, / tornati ai nostri fratelli e sorelle mammiferi, / e tutti sorvegliati da macchine di grazia amorevole." [Brautigan, 1967, trad. it. mia]

of a cybernetic forest
filled with pines and electronics
where deer stroll peacefully
past computers
as if they were flowers
with spinning blossoms.
I like to think
(it has to be!)
of a cybernetic ecology
where we are free of our labors
and joined back to nature,
returned to our mammal
brothers and sisters,
and all watched over
by machines of loving grace

Alla fine degli anni '60, alcune fasce della controcultura, come i *New Communalists* [Turner, 2006, p.33 trad. mia] – che in nome del “ritorno alla terra” fondavano comunità rurali e miravano a riformare la società americana attraverso la trasformazione delle coscienze individuali⁶ – abbracciarono esplicitamente gli ideali evocati dalla poesia di Brautigan: un’utopia tecnologica, “un’ecologia cibernetica”, in cui, grazie alle macchine e al loro sguardo amorevole – capace di liberare dalla schiavitù atavica del lavoro e di ricomporre dicotomie altrettanto antiche – uomo e natura, individuo e ambiente potessero finalmente riconnettersi. Tuttavia, in un’epoca solo di pochissimo anteriore, nello stesso *milieu* socioculturale, questa visione irenica delle macchine non era universalmente condivisa.

Sulle bluse, le camicie e le maglie degli studenti e dei dimostranti durante le manifestazioni del *Free Speech Movement* (FSM) nelle principali università californiane, come Berkeley, erano spillate schede perforate con la scritta “Sono uno studente: non piegare, infilzare o mutilare⁷. La formula, ripresa ironicamente dalle avvertenze stampate sulle *punched*

⁶ “I will call both those who actually established such communes and those who saw the transformation of consciousness as the basis for the reformation of American social structure New Communalists”.

⁷ “I am a student: do not fold, spindle, or mutilate”. La storia culturale di questo motto, così come quella delle schede perforate – simbolo e al tempo stesso strumento della necessità di un’implementazione fisica dell’informazione da dare in pasto ai sistemi di computazione – come vedremo in seguito – si dipana dal primo

cards IBM utilizzate per la registrazione degli studenti sin dagli anni '30, divenne un simbolo di resistenza contro l'alienazione e la burocrazia delle istituzioni accademiche americane [Heirich, 1968, p. 217; Lobar, 1992].

A mezzogiorno del 2 dicembre 1964, la piazza di fronte alla *Sproul Hall* dell'Università di Berkeley si riempì con oltre cinquemila studenti, seduti sul pavimento per ascoltare il leader del movimento Mario Savio (1942-1996). Armato di microfono, Savio arringò la folla definendo l'università una "autocrazia"; descrisse la sua dirigenza come un "consiglio di amministrazione" e il presidente come un "manager" [Savio, 1964], adottando una retorica che associava la disumanizzazione e la spersonalizzazione al mondo aziendale, quello stesso di cui, dopo il college, gli studenti sarebbero diventati membri. In questo quadro, studenti e docenti non erano altro che semplici ingranaggi di un'industria burocratizzata e alienante [Lubar, 1992; Turner, 2006, p. 11]. Le parole di Savio evocavano un immaginario da seconda rivoluzione industriale, in cui i corpi, ridotti a "materia prima" [Savio, 1964], manovrano leve e ingranaggi di una fabbrica. Poco prima che lo avvisassero dell'arrivo della cantautrice Joan Baez al presidio, Savio pronunciò parole destinate a diventare celebri:

C'è un momento in cui il funzionamento della macchina diventa così odioso, ti fa stare così male, che dovete mettere i vostri corpi sugli ingranaggi e sulle ruote... e dovete farla fermare. E dovete far capire alle persone che la gestiscono, alle persone che la possiedono, che se non siete liberi, la macchina non funzionerà affatto [Savio, 1964 trad. mia]⁸.

Ma forse, già allora, si colse che quelle parole rimandavano anche a un significato ulteriore. L'università, specchio dell'intera configurazione politica ed economica americana – e più in generale occidentale – della Guerra Fredda, stava diventando sempre più una fabbrica della conoscenza [Lubar, 1992], in un contesto in cui l'intero assetto sociale si ridefiniva intorno al calcolatore, secondo una visione organica che vedeva entità artificiali e biologiche come processori di informazione, i cui dati potevano essere ridotti a una scheda Hollerith.

impiego delle schede Hollerith (poi IBM) per il censimento federale statunitense del 1890 fino agli usi successivi nella cultura, nella controcultura e nella pop culture. Un'analisi dettagliata si trova in Lubar, 1992, pp. 43-55.

⁸ "There's a time when the operation of the machine becomes so odious, it makes you so sick at heart, that..you've got to put your bodies upon the gears and upon wheels.. and you've got to make it stop. And you've got to indicate to the people who run it, to the people who own it, that unless you're free, the machine will be prevented from working at all"

Le parole del discorso di Savio, allora, avevano per obiettivo la forma che aveva assunto il mondo contro cui egli e gli altri studenti e militanti del FSM lottavano: una società, burocraticamente strutturata, dell'informazione⁹.

Questa società aveva, secondo gli studenti delle università, per obiettivo la riduzione della persona in un'entità bidimensionale, come una scheda IBM, da mettere al servizio dell'apparato militare-industriale dell'America della Guerra Fredda. La "macchina". Al servizio di questa prima macchina, c'erano l'università come "macchina della conoscenza", e la macchina che effettivamente computava le informazioni ottenute astraendo dalla ricchezza dell'individualità, della profondità e molteplicità delle caratteristiche umane e incorporate in una scheda: il computer [Turner, 2006, p.12]¹⁰. Questi livelli concentrici, e reciprocamente specchio uno dell'altro, rappresentavano per i presenti in piazza a Berkeley il 2 dicembre '64, minacce di alienazione alla loro stesso vissuto, alla loro esperienza personale unica e irriducibile. Non è, quindi, da trascurare il riferimento che Savio, nel suo discorso, fa del *corpo*. Il corpo opposto come strumento di resistenza alla *macchina*¹¹.

Il corpo nella sua dimensione materiale, di carne, diviene l'istanza principale di ribellione al dominio dell'astrazione, della de-individualizzazione e della reificazione simboleggiati dalla scheda perforata.

Simbolicamente, nella manifestazione di fronte alla Sprout Hall, e nelle altre, l'occupare col proprio corpo i palazzi della macchina della conoscenza, il *sit-in*, divenne particolarmente importante. Sdraiarsi sui gradini dell'università, "gigantesca macchina per l'astrazione degli individui in materia prima informativa per l'industria della conoscenza", diventava un modo per riaffermare la propria umanità [Ivi, pp.12-13]¹².

⁹ "Savio's speech is famous, but few have realized that "the machine" he had in mind was not merely a mechanical metaphor for society; it was, at least as much, a metaphor for information technology" Lubar, 1992..

¹⁰ "To the protestors, the university was both a knowledge factory in its own right and a microcosm of the rigid, highly rationalized military-industrial complex it served. In that sense it modeled the hierarchical world of Cold War corporate adulthood for which many feared they were being trained. And at the time, no machine more commonly represented this stratified, depersonalized social order than the computer". Vedremo più avanti i risvolti sociologici dell'impiego della metafora cibernetica

¹¹ È impossibile non notare come nel porre queste dicotomie, si ricada nel dualismo, chiaramente di matrice cartesiana, tra *hardware* e *software* che struttura il *Funzionalismo*, filosofia della mente e modello esplicativo proprio delle teorie ispirate alla computazione e all'Intelligenza Artificiale, che incontreremo, analizzandolo meglio in seguito. Il Funzionalismo è "la teoria secondo cui gli stati mentali sono stati computazionali, esprimibili in termini di regole formali di manipolazione di simboli". Essi esprimono nient'altro che funzioni. Sono quindi implementabili da qualunque sistema, compresa una macchina digitale, che abbia le giuste caratteristiche. [Di Francesco, 2002, p.131.

¹² La coppia concettuale materialità/astrazione solleva una questione cruciale riguardo all'*informazione*, ovvero quale sia il suo statuto ontologico e se essa debba essere necessariamente implementata materialmente. Luciano Floridi ha denominato questo interrogativo come il "Problema di Wiener" [Floridi, 2005], rifacendosi a Norbert

L'episodio del 2 dicembre '64 è emblematico di una certa considerazione della "macchina", sia essa la tecnologia cibernetica figlia della Seconda Guerra mondiale e dispiegata nella Guerra Fredda e sia la società burocratizzata e gerarchizzata figlia del secondo dopoguerra.

Wiener che, nel suo classico *Cybernetics* affermava: "L'informazione è informazione, non materia o energia. Al giorno d'oggi, nessun materialismo che non ammetta questo può sopravvivere". [Wiener, 1948, p.177 trad. it.], Il concetto di informazione, con quello correlato di *entropia*, saranno analizzati durante tutto il corso di questo lavoro. La letteratura scientifica sembra concordare che non possa esistere "informazione senza rappresentazione". Questo concetto, unitamente al fatto che non esiste un dato puro, inteso come *dedomena*, "frattura nella struttura dell'essere", e che esso sia solo una *fictio* simile al noumeno kantiano o alla sostanza di Locke [Floridi, 2011, pp. 85-86 trad. it 2025], ha portato alcuni filosofi, come Chalmers, a sostenere la necessità che l'informazione debba avere un supporto materiale.: "Queste carte insegnano una lezione potente: l'informazione è fisica-o, almeno, l'informazione strutturale può essere fisicamente incarnata" [Chalmers, 2022, trad. it.]. Tuttavia, questa posizione è controversa, poiché non sembra seguire logicamente dalla necessità rappresentazionale dell'informazione [Floridi, 2005]. La questione della materialità dell'informazione riveste un'importanza decisiva anche nell'evoluzione storica delle tecnologie digitali, nel loro percorso verso la personalizzazione, nella *Human-Computer Interaction* (HCI) e nella forma che ha assunto la cultura digitale contemporanea. Il ruolo della corporeità, esemplificato dal *sit-in* del 2 dicembre 1964, trova il suo opposto speculare nella "*Declaration of the Independence of Cyberspace*", scritta da John Perry Barlow (1947-2019) e pubblicata su Internet l'8 febbraio 1996. Barlow, autore di testi per i *Grateful Dead*-band psichedelica simbolo della controcultura-e successivamente giornalista, saggista e tecnologo, compose la Dichiarazione mentre partecipava al *World Economic Forum* a Davos, in Svizzera, evento simbolo del capitalismo e della tecnocrazia mondiale. Durante la sua permanenza nella cittadina elvetica, osservò il Congresso americano approvare il *Telecommunications Act*, che includeva una clausola chiamata *Communications Decency Act*, mirata a limitare la pornografia su Internet. La Dichiarazione di Barlow fu la sua reazione contro quella che considerava un'intollerabile censura della libertà di espressione. "Governi del mondo industriale, stanchi giganti di carne e acciaio, vengo dal Cyberspazio, la nuova casa della mente. I vostri concetti legali di proprietà, espressione, identità, movimento e contesto non si applicano a noi. Sono basati sulla materia" [Barlow, 1996 trad. it. mia]. La materialità dei governi è la base della loro sovranità, ma questo potere non si applica nel mondo immateriale dell'informazione digitale. La mente, dal punto di vista informazionale, trova casa nella nuova frontiera immateriale, dove non c'è coercizione da parte di stati e governi. L'informazione è libera in quello spazio che William Gibson (1948- vivente), nel suo romanzo capolavoro del cyberpunk pubblicato il 1984, *Il Neuromante*, ha definito *Cyberspazio*. Senza indugiare oltre nell'analisi, è necessario notare come il corpo, la materia organica, sia il luogo in cui si esercita il potere. Il *sit-in* viene rovesciato in negativo. Se non c'è corpo non può esserci costrizione: "Non c'è niente qui. Le nostre identità non hanno corpo, quindi, diversamente da voi, non possiamo ottenere l'ordine con la costrizione fisica" [Barlow, 1996 trad. it. mia]. La posizione di Barlow è emblematica del modo in cui la destra libertaria della *new economy* ha invaso la nuova frontiera digitale in nome della deregolamentazione [Cfr. Turner, 2006, p. 7] e di come le posizioni che rappresenta, di cui è emblematico il motto "Information wants to be free", pronunciato da Stewart Brand, figura chiave, su cui ritorneremo, del ponte tra mondo controculturale e mondo *cyber*, abbiano plasmato il nostro *habitat* socio-culturale odierno. A questo proposito, si segnala la critica proposta da Jaron Lanier (1960- vivente), pioniere della realtà virtuale (VR), nel suo testo ormai classico di critica alla cybercultura, *You Are Not a Gadget: A Manifesto*. Lanier afferma: «L'informazione è esperienza alienata» [Lanier, 2010, p. 40 trad. it.]. La sua idea di realtà virtuale doveva essere un modo per reagire all'astrazione imposta dalla forma della codificazione esemplificata dal lavoro di Grace Hopper. La VR, invece, si presenta come un modo di manipolazione del codice sorgente mediante la reintroduzione del corpo, sebbene mediato dall'*avatar* e dalle tecnologie virtuali. [Cfr. Lanier, 2017, pp.364-365 trad. it.]. La corporeità ha un ruolo chiave anche nel percorso che ha portato alla forma attuale dei dispositivi digitali. Il pioniere dell'HCI e inventore del mouse, Douglas Engelbart, autore dell'archetipo del personal computer, l'*NLS* (on-line system) negli anni '60, è stato colui che ha impresso questa decisiva svolta verso la personalizzazione e l'*augmenting* del soggetto che utilizza lo strumento tecnologico. Come evidenziato da Bardini (2000): "Il modo in cui questo processo si è sviluppato ha avuto conseguenze significative per lo sviluppo del personal computer. La conseguenza più significativa è stata l'inclusione da parte di Douglas Engelbart del corpo dell'utente nell'interazione tra computer e i loro utenti" [Bardini, 2000, pp. 33-34 trad. it. mia].

Le immagini evocate dal discorso pubblico del leader del FSM, Mario Savio, si affiancano a quelle ben più famose e diffuse presenti nel romanzo di culto dello scrittore e “guru” della controcultura Ken Kesey (1935-2001), *Qualcuno volò sul nido del cuculo* [Kesey, 1962].

b. The Combine strikes again

La narrazione della lotta tra l'indomito e fiero piccolo criminale McMurphy (interpretato in modo indimenticabile da Jack Nicholson nella versione cinematografica del 1975 diretta da Miloš Forman) e l'inflessibile capo infermiera, Miss Ratched, dell'ospedale psichiatrico in cui è rinchiuso, è affidata al pellerossa Capo Bromden.

Quest'ultimo, come tutti i nativi americani, simboleggiava, per gli esponenti della controcultura americana, una forma di libertà più autentica e in armonia con la natura da cui trarre esempio¹³. Nel corso della narrazione, Bromden ha spesso visioni che svelano la vera natura del sistema rappresentato dall'ospedale psichiatrico, diretto con freddezza e spietata efficienza burocratica dalla Grande Infermiera, Miss Ratched.

“Ha riempito la borsa con i mille aggeggi che si propone di adoperare mentre è di turno oggi: rotelline e ingranaggi, ruote dentate così lucide da mandare duri riflessi, minuscole pillole che scintillano come porcellana, aghi, forcipi, pinze da orologiaio, rotoli di filo di rame” [Kesey, 1962, p.13, trad. it.]. Gli strumenti con cui l'infermiera esercita il suo potere rivelano il loro carattere meccanico. Le terapie rivelano la loro essenza tecnologica, esercitando coercizione sui pazienti:

Io mi allontanai, una volta, tenendo sotto la lingua una delle stesse capsule rosse, fingendo di averla inghiottita, e più tardi la schiacciai e l'aprii nel ripostiglio delle scope. Per una frazione di secondo, prima che tutto si tramutasse in polvere bianca, vidi che si trattava di un circuito elettronico in miniatura, come quelli alla cui costruzione collaboravo sotto le armi nel reparto Radar, fili e collegamenti microscopici, transistor, ma questo era stato progettato per dissolversi al contatto dell'aria [Kesey, 1962, pp.53-54 trad. it.].

La stessa struttura dell'ospedale è in realtà di tipo macchinico, fatta di ingranaggi, meccanismi e circuiti, nascosta appena sotto la superficie: “L'infermiere nero sbatté di piatto contro il muro e vi si appiccicò contro, poi scivolò giù sul pavimento come se la parete fosse

¹³ Ritroveremo più avanti la giustificazione “cibernetica” di questa preferenza per le piccole comunità, anche tribali.

stata lubrificata. Sentii valvole saltare e cortocircuitarsi dappertutto dentro il muro” [Kesey, 1962, p. 79 trad. it.]. Non a caso, la stanza più temuta di tutto l’ospedale, quella in cui si pratica l’Elettroshock – la cibernetica e la teoria dell’informazione hanno una chiara origine nello studio dei fenomeni elettrici - è chiamata dai pazienti “l’Officina dello Shock” [Kesey, 1962, p.102 trad. it.]¹⁴

La lotta è tra il sistema, la Cricca [*the Combine*]¹⁵ di cui fa parte la Grande Infermiera e il residuo di personale individualità e di libertà rimasto nei pazienti, dentro e fuori l’ospedale: “la

¹⁴ La visione della mente – un filo rosso che ritroveremo in tutto il corso di questo lavoro - che il romanzo di Kesey prende di mira, senza soffermarsi sugli aspetti concreti della pratica psichiatrica e della letteratura medica allora in uso negli ospedali psichiatrici, era quella che concepiva il cervello come un insieme di elementi discreti - i neuroni - considerati una delle possibili realizzazioni fisiche di strutture logiche binarie (0/1, on/off), le stesse che potevano essere implementate in qualsiasi relè o circuito elettronico. Segno del legame fra questa concezione organica e funzionale del cervello e la riflessione sulla malattia sono le parole di una figura chiave del movimento cibernetico, che incontreremo più volte in questo lavoro, Warren McCulloch, il quale in un articolo descrive esplicitamente il cervello come un calcolatore digitale. scrive: “Giudicando in base alla velocità di risposta, i neuroni rientrano nella fascia intermedia rispetto ai relè costruiti dall’uomo [...]. I neuroni, economici e abbondanti, sono anche multi-griglia; i più grandi possiedono migliaia di terminazioni, così da somigliare a transistor in cui la configurazione degli elettrodi determina il passaggio dei segnali. Inoltre, poiché i neuroni sono meccanicamente piuttosto stabili e si autoriparano, hanno una lunga vita utile. Non ne riceviamo di nuovi dal giorno in cui nasciamo fino a quello in cui moriamo. Ogni neurone nella nostra testa è vecchio quanto noi, e la maggior parte di essi è ancora funzionante. Naturalmente, possono essere uccisi da lesioni, incidenti, infezioni, mancanza di zucchero o ossigeno, o insufficiente apporto di sangue. Quando questo accade a un numero sufficiente di neuroni nei punti critici, soffriamo di malattie neurologiche e di ciò che chiamiamo psicosi organiche, indicando che il danno può essere rilevato nel cervello morto” [“Judging by speed of response, neurons fall into the middle range of man-made relays [...] Neurons, cheap and plentiful, are also multidigitated, the largest having upon them thousands of terminations, so that they resemble transistors in which the configuration of electrodes determines the gating of signals. Moreover, since neurons are mechanically rather stable and repair themselves, they have a long effective life. We get no new ones from the day we are born until we die. Every neuron in our head is as old as we are, and most of them are still serviceable. Of course, they may be killed by injury, accident, germs, want of sugar or of oxygen, or failure of their blood supply. When this happens to enough of them in the proper places, we suffer neurological diseases and what we call “organic” psychoses, meaning that the damage can be found in the dead brain”. McCulloch, Pfeiffer, 1949, trad. mia]. Per l’interesse del movimento cibernetico verso la psichiatria si veda Heims, 1991.pp.131-206. Claude Shannon, un altro dei protagonisti del nostro percorso, durante l’ottava Macy Conference, gli incontri interdisciplinari del movimento cibernetico, presentò una sua invenzione basata sulle sue teorie [Shannon, 1953]: il “Topo”, una macchina in grado di apprendere a navigare in un labirinto tramite memorizzazione di percorsi esplorati. Funzionava registrando due bit di informazione per ogni quadrato attraversato, sufficienti a determinare la direzione corretta. Tuttavia, in certi casi si bloccava in cicli infiniti, come A-B-C-D, mostrando un comportamento che Shannon definì una “*singing condition*”. “Una nevrosi!” esclamò Ralph Gerard. Shannon aggiunse un “circuito antinevrotico”: un contatore, impostato in modo da interrompere il ciclo quando la macchina ripeteva la stessa sequenza sei volte. Per Leonard Savage era un po’ barare: “Non ha un modo per capire che è ‘psicopatico’ – capisce solo che è andato avanti troppo a lungo?”. Shannon era d’accordo. “È tutto così umano,” osservò Lawrence K. Frank. “Avrebbe dovuto vederlo George Orwell,” disse Henry Brosin, uno psichiatra [Gleick, 2011, p. 230 trad. it.]. Shannon, 1953, in Pias, 2025, pp. 474-479]. In Wiener “la consapevolezza che cervello e macchina calcolatrice hanno molto in comune può suggerire nuovi e validi approcci alla psicopatologia, nonché alla stessa psichiatria” [Wiener, 1948, p. 191 trad. it.] lo spinse a dedicare un intero capitolo, il VII, di *Cybernetics* al tema di *Cibernetica e Psicopatologia*. Qui Wiener mette a confronto i guasti delle macchine calcolatrici con quelli del cervello, considerando i disturbi mentali come disordini della memoria, dell’informazione circolante e del traffico neuronico [Cfr. Wiener, 1948, pp. 191-203].

¹⁵ Kesey afferma chiaramente che l’obiettivo della «cricca» nell’ospedale psichiatrico sia quello di trasformare i pazienti in automi controllabili attraverso il potere dell’automazione tecnologica. Emblematica è la descrizione della Grande Infermiera che, “si alza da dietro la finestra, va in fondo alla sala infermiere, di fronte al pannello d’acciaio con manopole e pulsanti, inserisce una sorta di pilota automatico per far funzionare ogni cosa durante la

Cricca, l'enorme organizzazione, vale a dire un'enorme organizzazione il cui scopo è quello di adattare l'Esterno bene quanto ella ha adattato l'Interno". "Sì. Questo io so" afferma Capo Bromden: "La corsia è una fabbrica della Cricca. Serve a correggere gli errori commessi nei quartieri e nelle scuole e nelle chiese, a questo serve l'ospedale" [Kesey, 1962, p. 61, trad. it.]. Kesey pone la battaglia tra McMurphy e la Grande Infermiera come identica a quella tra i pellerossa di Warm Springs e la diga di Dalles sul fiume Columbia in Oregon [Cfr. Turner, 2006, p. 60], e Bromden, emblema dell'autenticità in quanto nativo, ne ha una chiara visione: "Cose pazzesche come un grande locale macchine giù nelle viscere di una diga, ove la gente viene squartata da operai-robot, non esistono. Ma se non esistono, come può un uomo vederle?" [Kesey, 1962, p. 131, trad. it.].

Anche in questo caso la "macchina" esercita il suo potere reificante e oppressivo sulla materialità di chi, o cosa, si venga a trovare tra i suoi ingranaggi e circuiti.

Ma come è possibile essere passati dalla visione della poesia di Brautigan della spaventosa minaccia di spersonalizzazione evocata dalle immagini del romanzo di Ken Kesey e dalle parole del leader del FSM Savio? Da dove giungono pacifiche immagini di un'utopia cibernetica, dove addirittura la tecnologia diviene strumento di una piena realizzazione del Sé in armonia con la natura?

Le immagini del rapporto tra uomo, mondo, società e sviluppo scientifico che emergono da questi esempi sembrano prefigurare il dibattito contemporaneo intorno al digitale e alle *Information and Communication Technologies* (ICT), polarizzato tra chi sostiene ne ha una visione messianica e chi, al contrario, profetizza un'apocalisse tecnologica.

Da un lato si delineano visioni di un futuro in cui la tecnologia, quasi redentrice, promette di liberare l'uomo dai suoi mali atavici - l'invecchiamento, la malattia, la fatica del lavoro -, di potenziarne al massimo le capacità e di ricucirne il rapporto con la natura. Dall'altro emergono le ombre di un domani in cui lo sviluppo tecnologico diventa sinonimo di alienazione e distruzione ambientale, con effetti disruptive che condurrebbero a una disoccupazione di

sua assenza e viene nella sala comune portando il registro e un cestino pieno di appunti" [Kesey, 1963, p.64 trad. it.]. La critica di Kesey all'automazione è esplicita. Raccontando di un paziente che, prima del ricovero, aveva un lavoro tranquillo nelle ferrovie, l'autore sottolinea come "non fece mai installare alcun controllo automatico" e considera questa una vittoria ottenuta «grazie alla forza di volontà e al coraggio che non erano mai riusciti a schiacciargli nella testa» [Ivi]. La metafora della produzione industriale di automi trova un corrispettivo nella condizione dei pazienti cronici, considerati come scarti di questa fabbrica. I Cronici "non rimangono in ospedale per essere guariti, ma soltanto per impedire che si aggirino nelle strade dando una cattiva nomea al prodotto. I Cronici sono qui per sempre, il personale lo ammette" [Ivi]. È difficile non notare le affinità tra la visione critica di Kesey e le posizioni espresse da intellettuali come Michel Foucault o psichiatri come Franco Basaglia, che hanno duramente criticato la psichiatria tradizionale e i suoi metodi.

massa e priverebbero l'uomo delle qualità che lo rendono umano, come il pensiero critico e la creatività, fino a spingersi a scenari fantascientifici in cui macchine dominanti riducono in schiavitù l'intera razza umana.

La complessità scientifica, tecnica e culturale della questione, tratteggiata dagli esempi precedenti e le cui conseguenze si dipanano fino a dare forma alla nostra contemporaneità - un'epoca segnata dalla pervasività della computazione, del digitale e delle ICT, l'età della *Quarta Rivoluzione* [Floridi, 2014] - richiede un ulteriore livello di analisi. È necessario, infatti, tornare ad alcuni episodi della genesi scientifica e culturale della cibernetica, che mettono in luce la visione del mondo e dell'uomo emersa sin dai primi vagiti di queste tecnologie, con tutto il loro potenziale rivoluzionario e trasformativo rispetto all'identità umana, ai rapporti intersoggettivi e al legame con l'ambiente circostante. Una tensione che, su sponde opposte, Brautigan, Kesey e Corso avevano già intuito.

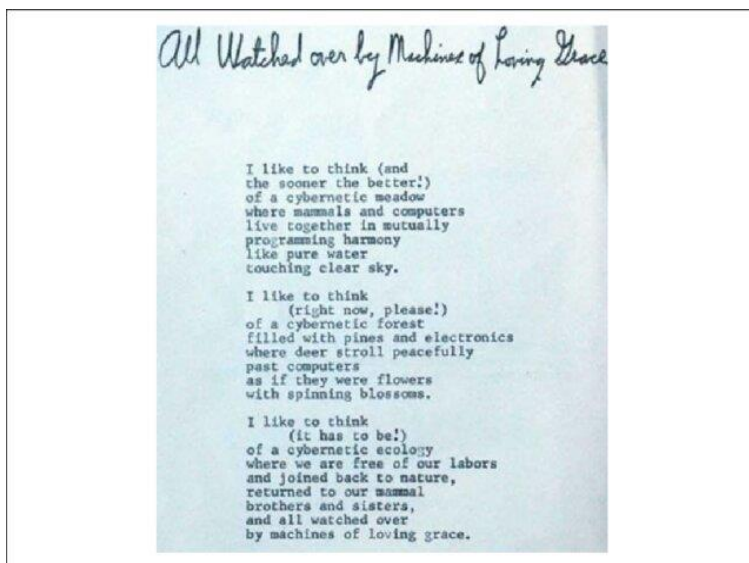


Fig.1 "All Watched Over by Machines of Loving Grace." fonte: Brautigan [1967]

2. Oggi il predittore, domani il mondo: genesi ed eredità della visione del mondo cibernetica

a. Macchine perturbanti

Un ventennio prima, il 1° luglio del 1942, circa sei mesi dopo l'entrata degli Stati Uniti nella Seconda Guerra Mondiale seguito all'attacco giapponese a Pearl Harbor del 7 dicembre 1941, George R. Stibitz (1904-1995), pioniere dei calcolatori digitali e ricercatore presso i Bell Labs, nonché direttore della sezione dell'NDRC presso la quale prestava servizio Norbert Wiener (1894-1964), fece visita al futuro padre della cibernetica nel suo "piccolo laboratorio" situato nella stanza 2-244 dell'edificio 2 del MIT [Wiener, 1953, p.248].

Questo laboratorio era adiacente all'edificio 6, dove, su una casupola di legno sul tetto, il team del *Radiation Laboratory* del MIT aveva iniziato il suo lavoro nel gennaio del 1941 [Conway, Siegelman, 2005, pp.169-170 trad. it.]. L'esito della visita spinse Stibitz a registrare sul suo diario di lavoro la sensazione che aveva provato osservando il lavoro che Wiener e l'ingegnere Julian Bigelow [1913-2003]. Questa sensazione era di sbalordito e perturbante stupore [*uncanny*], rispetto a quanto Wiener stava realizzando e ai suoi risultati: un predittore antiaereo, la cui realizzazione era la sua "personale sfida bellica" [Conway, Siegelman, 2005, p. 157 trad. it.].

Il problema affrontato da Wiener, e che desiderava risolvere con l'aiuto del suo predittore, era quello di "indagare su come sia possibile, con vari metodi, predire segnali che non siano puramente lineari". Cercava, cioè, di sviluppare un sistema di calcolo capace di superare il metodo geometrico, applicabile solo a traiettorie semplici, per "investigare i problemi di predizione per segnali che contengono rumore dovuto al tracciamento o agli errori di volo, o per segnali corrispondenti a vari tipi generali di volo curvilineo" [Wiener, 1942 trad. mia]¹⁶.

I risultati ottenuti dalla macchina di Wiener, in termini di predizione, consentivano inizialmente previsioni per un massimo di due secondi. Tuttavia, il dispositivo presentava una caratteristica significativa che ne accresceva il valore nel contesto reale: lo schema elaborato da Wiener e Bigelow comprimeva il tempo di un fattore compreso tra quattro e cinque. In questo modo una predizione di due secondi corrispondeva, nel mondo reale, a una predizione

¹⁶ "Thus it is important to investigate how one can, by various methods, predict other than a pure linear signal. For analytical curves of varying amounts of complexity (horizontal circles, helices with dive or climb, etc.) the prediction procedures can be worked out by simple geometrical methods. But to investigate prediction problems for signals which contain noise due to tracking or to flight errors, or for signals corresponding to various general types of curved flight, that is a much deeper and more difficult matter".

di circa dieci secondi. Tale aspetto risultava particolarmente rilevante in battaglia, poiché un proiettile antiaereo impiegava almeno venti secondi per raggiungere il proprio obiettivo all'altitudine corrispondente. Il predittore appariva quindi promettente per migliorare l'efficacia delle difese antiaeree [Cfr. Galison, 1994, pp. 57-58 trad. it.].

Questi erano i risultati che avevano suscitato lo stupore di Stibitz. Il predittore statistico [fig. 2] di Wiener “compie miracoli”, sebbene Warren Weaver (1894-1979) – matematico, direttore della Sezione Scienze Naturali della *Rockefeller Foundation* e della sezione di matematica applicata del NDRC, nonché pioniere della teoria dell'informazione e della traduzione automatica – “non sia ancora convinto se questo sia un miracolo utile o inutile”. Ma lo stesso Weaver, impressionato a sua volta dalla performance del predittore, prosegue nell'annotazione di Stibitz: “minaccia di portarsi una sega da taglio alla prossima visita e tagliare le gambe del tavolo per vedere se non abbiano qualche filo nascosto da qualche parte”. E ancora: “Il comportamento dei loro [di Wiener e Bigelow] strumenti è certamente perturbante” [Stibitz, 1942 D, in Galison, 1994, p. 57, trad. it.].

A cosa era dovuta la sensazione di *Unheimlich*, di *uncanny*, di perturbante provata da Stibitz e Weaver alla vista di quella macchina, un insieme di metallo e circuiti che avrebbe potuto benissimo essere sostenuto da una robusta scrivania? Rispondere a questa domanda significa confrontarsi direttamente con la visione dell'uomo, del mondo e degli enti che lo popolano, nonché dei rapporti che li legano, propria del pensiero di Wiener. Significa anche dare ulteriore profondità alle questioni sollevate dagli esempi tratti dalla controcultura e offrire uno spaccato dell'influenza che i primi vagiti della nuova scienza hanno esercitato e tuttora esercitano. L'episodio narrato nel diario di Stibitz risale al luglio del 1942. Questo passaggio illumina non solo la definizione del rapporto tra essere umano, contesto e macchina, ma anche la configurazione assunta dalla cultura digitale contemporanea: l'importanza delle tecnologie dell'informazione, le modalità con cui esse esercitano la loro influenza e le trasformazioni che producono, e continuano a produrre, nel modo in cui diamo concettualmente forma a noi stessi e al mondo.

Wiener “aveva già iniziato a riflettere sulle più generali ramificazioni di questa specie di macchina” [Galison, 1994, p. 57, trad. it.] legandole ad una più ampia riflessione intorno alla natura del comportamento e della fisiologia degli organismi, mostrando quella tensione intellettuale che si manifestava nella sua inveterata tendenza alla generalizzazione delle conseguenze delle sue scoperte. “Ciò che Wiener era pronto a fare, anche nei giorni peggiori

della guerra, era di volgersi alle implicazioni psicologiche e filosofiche del predittore” [Galison, 1994, p. 60 trad. it].

Il predittore antiaereo sviluppato da Wiener mirava a risolvere le difficoltà poste dalle traiettorie non lineari degli aerei in volo. Pur avendo fornito alcuni risultati promettenti - tali da attirare l'attenzione di figure come Stibitz e Weaver - il dispositivo faticava a raggiungere una piena efficacia. In un rapporto indirizzato a Weaver il 15 gennaio 1943, a pochi mesi di distanza dalla relazione *Statistical Methods of Prediction in Fire Control* [Wiener, 1942], in cui aveva illustrato il suo sistema antiaereo, Wiener ammetteva che il metodo non aveva ancora prodotto gli esiti attesi. Inoltre, la sua ricerca si trovava a competere con il sistema di predizione geometrica sviluppato da Henrik Bode ai Bell Labs, che godeva del vantaggio di poggiare su una tecnologia già consolidata [Wiener, 1942].

È quasi certo che il vantaggio matematico del mio metodo emergerà davvero per tempi di volo considerevolmente più brevi di 20 secondi. Qui, tuttavia [...] il vantaggio di uno strumento elaborato [come il suo predittore, limitato ad armi di grosso calibro] e non troppo facilmente trasportabile diventa piuttosto discutibile. In che misura il risultato negativo di questa indagine sia dovuto a un cattivo tracciamento, in che misura alla restrizione del passato utilizzabile a 10 secondi, e in che misura al fatto che l'aereo nemico ha una considerevole possibilità di cambiare il suo schema di volo, sia volontariamente che involontariamente, nei venti secondi di volo del proiettile, non è ancora del tutto chiaro [Wiener, 1942 trad. mia]¹⁷.

Il suo lavoro non stava dando risultati, non riusciva ad abbattere i nemici, anzi, il suo metodo era un modo per “mostrare semplicemente come non provare a ucciderli” [Galison, 1994, p.59 trad it.]. Ma su cosa si fondava questo metodo? Quali erano le sue peculiarità?

Il metodo elaborato da Wiener era basato sulla considerazione che alla base delle traiettorie complesse vi fosse la “capacità del nemico di manovrare” [Galison, 1994, p.59 trad it.]. Il fattore di instabilità che rendeva così difficile prevedere curve che era necessario calcolare per abbattere gli aerei nemici, altri non era che il pilota nemico: “Ci rendemmo conto

¹⁷ “It is almost certain that the mathematical advantage of my method will really appear for considerably shorter times of flight than 20 seconds. Here, however, [...] the advantage of an elaborate, not too easily portable instrument becomes quite questionable. To what extent the negative result of this investigation is due to bad tracking, to what extent [...] and to what extent to the fact that the enemy plane has a very considerable chance to change its flight pattern, whether voluntary or involuntary, in the twenty seconds of projectile flight, is not yet fully clear.”

che la “casualità” o l’irregolarità del percorso di un aeroplano è introdotta dal pilota” [Galison, 1994, p.49 trad it.].

Il pilota nel costante tentativo di correggere la traiettoria del velivolo in relazione al suo obiettivo si comporta come un “servomeccanismo”. La nozione di servomeccanismo che Wiener aveva in mente¹⁸ era quella espressa poi da L. A. MacColl in *Fundamental theory of servomechanisms* del 1945 [MacColl, 1945], dove ne veniva data la seguente definizione:

Il compito essenziale di un servomeccanismo è imporre al segnale di uscita $y(t)$ la stessa forma funzionale del segnale di ingresso $x(t)$ con la restrizione che l’energia associata a $y(t)$ debba essere ricavata da una sorgente locale e non fornita direttamente dal segnale di ingresso [...] Un servomeccanismo è un sistema che svolge la funzione essenziale di un servomeccanismo, come enunciata sopra, e che è costruito in modo tale che la causa efficace che mette in funzione il sistema sia proporzionale alla seguente combinazione dei segnali di ingresso e di uscita: $e(t) = Kx(t) - y(f)$. [MacColl, 1945, pp. 2-3 trad. mia]¹⁹.

E nel comportarsi di questo come tale, si verifica una variazione nella sua *cinestesia*²⁰: “La reazione cinestetica del pilota al movimento dell’aeroplano è molto differente da quella che i suoi altri sensi normalmente lo porterebbero ad aspettarsi, così che per il volo di precisione, egli deve dissociare la sua cinestetica dal suo senso della visione” [Galison, 1994, pp.49 trad. it.]. Strettamente correlata a questa intuizione metodologica, che sarà vista più da vicino a breve, vi era un’altra novità che caratterizzava il trattamento dei segnali e delle traiettorie

¹⁸ Fa notare Galison come Wiener approvasse e citasse spesso il lavoro di MacColl sui servomeccanismi [Galison, 1994, p. 49, trad. it.]. Qui questi dispositivi vengono identificati con sistemi “schiavizzati” [*slaved*] che impiegano il feedback [cfr. MacColl, 1945, pp. 1-9]. W. Weaver, nella sua introduzione al lavoro di MacColl, fa risalire l’origine dei servomeccanismi al lavoro di fine Settecento di James Watt sulle macchine a vapore [Weaver, 1945, p. vii]. Questa opinione sulla paternità del servomeccanismo è condivisa da Wiener [Wiener, 1948, p. 135]. In particolare, Watt introdusse nel 1788 il governatore centrifugo, due sfere attaccate ad un albero rotante, per regolare automaticamente l’afflusso di vapore, impedendo che la macchina variasse eccessivamente la propria velocità al variare del carico.

¹⁹ “The essential function of a servomechanism is to impose upon the output signal $y(t)$ the same functional form as the input signal $x(t)$ subject to the restriction that the energy associated with $y(t)$ shall be derived from a local source, and shall not be furnished directly by the input signal [...] A servomechanism is a system which performs the essential function of a servomechanism stated above, and which is so constructed that the effective cause which operates the system is proportional to the following combination of the input and output signals: $e(t) = Kx(t) - y(f)$.”

²⁰ Per cinestesia, o propriocezione, si intende «il flusso sensorio continuo ma inconscio proveniente dalle parti mobili del nostro corpo (muscoli, tendini, articolazioni), che ne controlla e ne adatta di continuo la posizione, il tono e il movimento, in un modo però che a noi rimane nascosto perché automatico e inconscio» (Sacks, 1985, p.39 trad. it). Il concetto, studiato per la prima volta in modo approfondito dal neurofisiologo, medico (e poeta) britannico C.S. (1857-1952) [Sherrington, 1906, pp. 130-132, pp. 335-343] indica quel *sesto senso* indispensabile per il senso di noi stessi. È ciò che ci fa avvertire il nostro corpo come nostro.

irregolari da parte di Wiener. Il suo metodo di predizione, e ne vedremo più avanti gli sviluppi, era di natura *statistica* [Wiener, 1942].

Usando questo metodo Wiener, Bigelow e e gli altri due membri del loro staff che componevano il loro staff, affrontavano un problema sia “fisico che fisiologico” [Galison, 1994, p.49 trad. it.]. Questo riguardava sia l'*hardware* di acciaio e circuiti di cui era fatto l'aereo e sia quello di carne e nervi del pilota, con quest'ultimo che “in mezzo alle esplosioni dell'artiglieria antiaerea, la turbolenza dell'aria e lo spazzare dei riflettori, cerca di guidare un aeroplano all'obiettivo” [Galison, 2004, p.49 trad. it.]. L'ipotesi alla base del metodo statistico di Wiener era che “l'attività umana sotto stress tende a prodursi un ripetitivamente e quindi in maniera predicibile” [Galison, 2004, p.49 trad. it.].

Bigelow, che era un pilota esperto ed appassionato²¹, aiutò Wiener a ricreare questa strettissima interrelazione tra uomo e macchina costruendo un apparato che andava a incidere sulle reazioni cinestetiche dell'ipotetico pilota in modo da individuare le modalità di correzione a retroazione operate da quest'ultimo. Questo strumento era fatto per “per generare curve non dissimili dalla traiettoria di un aeroplano come materiale grezzo per testare il metodo statistico” [Wiener, 1942]²².

L'apparato generatore di traiettorie veniva stabilizzato, in modo statistico, assegnando all'operatore, lo pseudo-pilota di cui si voleva studiare il comportamento in volo, “un compito chiaramente al di là della sua capacità di esecuzione accurata” [Wiener, 1942]²³.

Racconta Wiener: “Avevamo [lui e Bigelow] riflesso sul soffitto un punto di luce, che si muoveva in un percorso più o meno periodico. Abbiamo cercato di seguire questo punto con un altro, riflesso in uno specchio controllato da un certo apparato”. Il movimento effettivo del punto luce e quello della manovella, la cui rotazione avrebbe dovuto regolarlo, non erano in una relazione di proporzionalità “ma piuttosto a una mescolanza piuttosto complicata di derivate e integrali di questo”²⁴ [Wiener, 1953, p.250]²⁴. Vengono alla luce le due anime di Wiener, il

²¹ “He was an avid airplane pilot and flew regularly into his 80's, when he renovated a plane as a hobby”, Markoff, *New York Times*, 22 febbraio, 2003.

²² “Mr. Bigelow also designed an apparatus for generating curves not unlike the track of an airplane as raw material for testing the statistical method”.

²³ “This apparatus for generating “tracks” as statistically stabilized by giving the operator a task to perform which was strictly beyond his capacity for accurate performance.”.

²⁴ “The problem of generating an irregular curve with a statistically controllable degree of irregularity was quite interesting. We had reflected on the ceiling a spot of light, moving in a more or less periodic course. We tried to follow this spot with another, reflected in a mirror which was controlled by a certain apparatus. In this apparatus, the actual motion of the spot was not proportional to the turning of the crank which regulated it but to a rather complicated mixture of derivatives and integrals of this motion”.

filosofo²⁵, ma anche il matematico, l'esperto "dell'analisi armonica e del moto browniano come strumenti per lo studio dei fenomeni irregolari distribuiti nel tempo" [Wiener, 1953, p.250]²⁶.

All'interno dell'apparato, veniva mossa una luce guida sulla parete della stanza di test e l'operatore doveva mediante un controllo muovere una luce che seguiva la prima. L'operatore pilota "era fornito di una leva di controllo volutamente lenta che guidava la posizione e il moto di una seconda macchia di luce colorata. Il compito del pilota era di guidare la macchia colorata nell'"obiettivo" (Galison, 1994, p.50 trad. it.). Questo controllo portava "a un apparato che falsificava deliberatamente le sensazioni cinestetiche stimulate nel controllarlo" (Wiener, 1942)²⁷.

Il compito era reso intenzionalmente difficile dagli sperimentatori per "creare esattamente la dissociazione tra senso cinestetico e informazione visiva che il pilota doveva fronteggiare nel teatro di guerra e avrebbe quindi condotto allo stesso tipo di difficoltà di retroazione" [Galison, 1994, p.50 trad. it.]: "la manovella era collegata a un sistema di pesi e molle, che era molto lontano dal dare le sensazioni cinestetiche che si assocerebbero naturalmente a un apparato di questo tipo. In altre parole, il punto doveva essere mosso da un

25 Una leggenda della famiglia Wiener fa risalire la loro origine al filosofo ebraico e medico del dodicesimo secolo Moises Maimonides. [Conway, Siegelman, 2005, p. ix, p.9 trad. it.]. La vita di Wiener è stata sempre segnata da uno spiccato interesse per la filosofia, sin dagli anni della sua infanzia, quando assurse a popolarità nazionale come "bambino prodigio" [Wiener, 1953a, p.3]. Nell'autunno del 1906 un reporter della rivista di New York, *The World*, la testata di Joseph Pulitzer, fece un reportage sul piccolo genio figlio di Leo Wiener, slavista erudito e professore ad Harvard, il quale aveva deciso che suo figlio Norbert conoscesse le lingue, le scienze e la matematica molto prima di iniziare le scuole elementari. "Il bambino, Norbert, distingueva le lettere dell'alfabeto a diciotto mesi. Sotto la guida di suo padre aveva iniziato a leggere a tre anni, declamare in greco e latino a cinque e in tedesco subito dopo. A sette anni si era dedicato alla chimica, a nove ad algebra, geometria, trigonometria, fisica, botanica e zoologia; e in quell'autunno, a undici anni, dopo soli tre anni e mezzo di istruzione scolastica, era entrato al Tufts College della vicina città di Medford". Ma la vera passione del "bambino prodigio" era la filosofia. Incontrò il giornalista portandosi sotto al braccio *I problemi della filosofia* di Hibben, e alla domanda dello stupito reporter su come un bambino della sua età potesse preferire i filosofi alle favole, il piccolo Wiener rispose: "La filosofia è più interessante delle favole, tutto qui. In effetti, per me la filosofia è il paese delle fiabe». Proseguì la conversazione spiegando al sempre più attonito giornalista che la differenza tra un Omero e un filosofo dell'800 come Haeckel, l'inventore del termine "ecologia", era che il filosofo «cercava di risolvere l'enigma dell'universo. Omero si limitava a intessere storie" [Conway, Siegelman, 2005, pp.26-29 trad. it.] "Wiener era un filosofo e aveva facilità di elaborazione» disse di lui Bigelow in riferimento alla costruzione dell'apparato per generare traiettorie irregolari" [Conway, Siegelman, 2005, p.163 trad. it.]

26 "My war work on filtering and prediction of time series had represented an extension of my earlier work on generalized harmonic analysis and on the Brownian motion as tools for the study of irregular phenomena distributed in time". Per l'analisi matematica del lavoro di Wiener si rimanda al, a mio avviso fondamentale, Masani, 1990 e ai IV voll. dell'opera completa wieneriana, Wiener, 1976-1985., corredata da commenti di prestigiosissimi matematici.

27 "This control led on the one hand to an apparatus which deliberately falsified the kinesthetic sensations stimulated in controlling it, and on the other, controlled its spot of light in velocity rather than in position".

controllo che era complicato sin dall'inizio e, inoltre, sembrava completamente sbagliato" [Wiener, 1956, p.250]²⁸.

Le conseguenze, dal punto di vista metodologico, filosofico e antropologico, sono notevolissime: "Wiener e Bigelow considerarono l'artigliere, il cannone, l'aeroplano e il pilota [gli elementi riprodotti dallo strumento meccanico] come un sistema probabilistico integrato" [Dyson, 2012, p.128 trad. it.]. Era essenziale, al fine di poter prevedere il comportamento dei velivoli nemici, concettualizzare l'elemento umano (pilota e artigliere) come servomeccanismi «entro un sistema unico» [Galison, 1994, p.52 trad. it.] con quello meccanico. Il *blurring*, lo sfumare delle differenze tra dimensione biologica e quella artificiale (insieme ai metodi probabilistici per trattarlo), e l'idea che il comportamento di queste fosse un sistema di retroazione – elementi cardine della nascente disciplina – si erano già manifestati in modo embrionale nell'ipotesi di lavoro di Wiener, ben prima di dare un nome alla scienza – cibernetica, dal greco κυβερνήτης, "timoniere" - al nuovo campo interdisciplinare che scaturiva da questo tentativo²⁹.

Scrivi Wiener nella sua autobiografia a questo proposito:

Non sembra persino remotamente possibile eliminare l'elemento umano per come si mostra esso stesso nel comportamento del nemico. Perciò, al fine di ottenere un trattamento matematico quanto più completo possibile del problema del controllo totale, è necessario assimilare le parti differenti del sistema a una singola base, umana o meccanica. Poiché la nostra comprensione degli elementi meccanici del puntamento di un cannone ci è parsa essere molto più avanzata della nostra comprensione psicologica, abbiamo scelto di provare a trovare un analogo meccanico del puntatore del cannone e del pilota dell'aeroplano [Wiener, 1956, p.251-252]³⁰.

La trattazione matematica del problema della predizione, e della variabile di incertezza identificata da Wiener e Bigelow nel pilota, poteva essere affrontata solo riducendo i singoli elementi che componevano il sistema da studiare a una base comune, riproducibile

²⁸ "Moreover, the crank was attached to a system of weights and springs, which was very far from giving the kinesthetic sensations which one would naturally associate with an apparatus of the sort. In other words, the spot had to be moved by a control which was complicated to begin with and, furthermore, felt completely wrong".

²⁹ Per la genesi della cibernetica dal lavoro sul predittore antiaereo si veda Masani, Phillips, 1985.

³⁰ "However, it does not seem even remotely possible to eliminate the human element as far as it shows itself in enemy behavior. Therefore, in order to obtain as complete a mathematical treatment as possible of the over-all control problem, it is necessary to assimilate the different parts of the system to a single basis, either human or mechanical. Since our understanding of the mechanical elements of gun pointing appeared to us to be far ahead of our psychological understanding, we chose to try to find a mechanical analogue of the gun pointer and the airplane pilot".

meccanicamente. Quando il pilota di un veicolo è impegnato in manovre evasive in mezzo al fuoco nemico, prosegue Wiener, “il suo schema di volo è strettamente legato non solo ai limiti del suo aereo ma anche a quelli del suo sistema nervoso, così che la sua azione non è troppo diversa da quella dell'azione umana ipotetica che avevamo progettato”. Ma questo non vale solo per il pilota: anche l'artigliere impegnato a colpirlo “introduce certi errori causali a causa dei limiti dei suoi organi sensoriali e dei suoi muscoli” [Wiener, 1956, p.251 trad. mia]. Entrambi gli elementi umani, conclude Wiener, “si combinano come parte dei processi semimeccanici con cui l'artigliere antiaereo abbatte il suo bersaglio” [Wiener, 1956, p.251 trad. mia]³¹.

Il comportamento di questi servomeccanismi integrati è quello, per cui “osservando gli errori commessi in un certo schema di comportamento e contrastando questi errori con azioni deliberate volte a ridurli” [Wiener, 1956, p.252 trad. mia]³². Sono cioè, *meccanismi di retroazione*, di *feedback*, che correggono via via la performance in base ai risultati che di volta in volta ottengono in relazione ad uno scopo.

Si comprende ora meglio l'accento posto sulla propriocezione del pilota. A questo proposito, noterà in seguito Wiener nel suo classico del '48 – a ribadire l'importanza del concetto – come, per compiere qualunque azione, anche le più elementari, ad esempio l'afferrare qualcosa con la mano, nessuno, tranne degli esperti anatomisti, sa esattamente quali muscoli siano coinvolti; e anche tra questi, “ve ne sono pochi, se non nessuno, che possano eseguire questa azione mediante una cosciente e intenzionale successione di contrazione dei muscoli implicati” [Wiener, 1948, p.30 trad. it.]. Ciò che vogliamo fare, il nostro fine, è agguantare l'oggetto. Una volta optato per lo scopo – l'afferrare la cosa – questo obiettivo rimane la parte cosciente della nostra azione, mentre “il nostro movimento procede in modo tale che [...] quanto manca alla presa della matita decresca progressivamente. Questa parte dell'azione non si svolge a livello di piena consapevolezza”. Come si giunge allora all'obiettivo? “Occorre che al sistema nervoso pervenga ad ogni istante, consapevolmente o meno, una segnalazione dell'ammontare di quanto ancora manca” [Wiener, 1948, p.30 trad. it.] al

³¹ “On the one hand, when the airplane pilot is flying and taking evasive action, his pattern of flight has a great deal to do with not only the limitations of his plane but those of his nervous system, so that his action is not too different from that of the hypothetical human action we had designed. On the other hand, the anti-aircraft gunner uses a technique in which he cannot follow his target perfectly but in which he introduces certain random errors because of the limitations of his sense organs and muscles. These two sorts of human elements are combined as part of the semimechanical processes by which the anti-aircraft gunner brings down his target”.

³² “In both these cases, the operators seemed to regulate their conduct by observing the errors committed in a certain pattern of behavior and by opposing these errors by actions deliberately tending to reduce them. This method of control appeared to us not unlike a method already known in electric circuits and now being applied in servomechanisms, or systems by which we switch in an outside source of power for control purposes, such as occurs in the power steering of a truck”.

raggiungimento dello scopo. È dunque necessario il senso cinestetico/proprioceettivo, che permette di correggere via via il comportamento finalizzato a livello della stessa struttura fisiologica di chi compie l'azione.

Tornando ai piloti e agli artiglieri, il loro comportamento non è libero. Non possono operare come vogliono. Quando un pilota manovra un apparecchio che a velocità elevatissime cerca di evitare il fuoco nemico “il suo schema di volo è strettamente legato non solo ai limiti del suo aereo ma anche a quelli del suo sistema nervoso”. Lo stesso vale per l'artigliere che non può seguire perfettamente il suo bersaglio, ma “introduce certi errori casuali a causa dei limiti dei suoi organi sensoriali e dei suoi muscoli” [Wiener, 1956, p.251]³³.

Lo schema d'azione dell'artigliere che punta e fa fuoco verso il velivolo nemico e quello del pilota che cerca di evitare di essere abbattuto sono “due tipi di elementi umani si combinano come parte dei processi semimeccanici con cui l'artigliere antiaereo abbatte il suo bersaglio”. [Wiener, 1956, p.251]³⁴. Il comportamento orientato a uno scopo dei due elementi umani integrati nel sistema probabilistico del predittore [Fig.2] è deliberatamente volto a ridurre l'errore rispetto all'obiettivo. Come osserva Wiener: “Questo metodo di controllo ci è sembrato non dissimile da un metodo già noto nei circuiti elettrici e ora applicato nei servomeccanismi, o sistemi in cui utilizziamo una fonte di potenza esterna per scopi di controllo, come avviene nel servosterzo di un camion. Chiamiamo questo *feedback* negativo” [Wiener, 1956, p.251]³⁵.

³³ “The anti-aircraft gunner uses a technique in which he cannot follow his target perfectly but in which he introduces certain random errors because of the limitations of his sense organs and muscles”.

³⁴ “These two sorts of human elements are combined as part of the semimechanical processes by which the anti-aircraft gunner brings down his target”.

³⁵ “This method of control appeared to us not unlike a method already known in electric circuits and now being applied in servomechanisms, or systems by which we switch in an outside source of power for control purposes, such as occurs in the power steering of a truck. We call this negative feedback”. Al lavoro sul predittore, nato nelle esigenze del secondo conflitto mondiale, si deve anche la nascita del concetto di Cyborg, dell'organismo cibernetico, in cui le differenze tra parti biologiche e componenti artificiali, svaniscono.” La cibernetica, l'intelligenza artificiale e la psicologia cognitiva hanno fatto ampio affidamento sui computer come metafore e modelli della mente, concepita come un sistema capace di risolvere problemi, di autocontrollarsi e di elaborare simboli. La parola “cyborg”, o organismo cibernetico, cattura lo sfumare strategico dei confini insito in queste metafore. Il discorso sul cyborg, costruendo tanto le menti umane quanto le intelligenze artificiali come macchine dell'informazione, contribuì a integrare le persone nei complessi sistemi tecnologici” [“Cybernetics, AI, and cognitive psychology relied crucially upon computers as metaphors and models for minds conceived as problem-solving, self-controlling, symbol-processing systems. The word “cyborg,” or cybernetic organism, captures the strategic blurring of boundaries inherent in these metaphors. Cyborg discourse, by constructing both human minds and artificial intelligences as information machines, helped to integrate people into complex technological systems”, Edwards, 1996, p.2]. Per un'analisi del concetto di Cyborg e dei suoi rapporti con la cibernetica, nonché con la dimensione sociotecnica e politica in cui è emerso (con le ricadute nell'immaginario del discorso pubblico), si rimanda a Edwards, 1996, e Kline, 2017. Un'analisi filosofica, con ricadute sulla nascita del cosiddetto “post-moderno”, degli organismi cibernetici è in Galison, 1994, pp.74-76.

Come nota Peter Galison, col riconoscimento della centralità del feedback e di una comune base umano-meccanica: “La teoria servomeccanica sarebbe diventata la misura del mondo”. [Galison, 1994, p. 52 trad. it.]

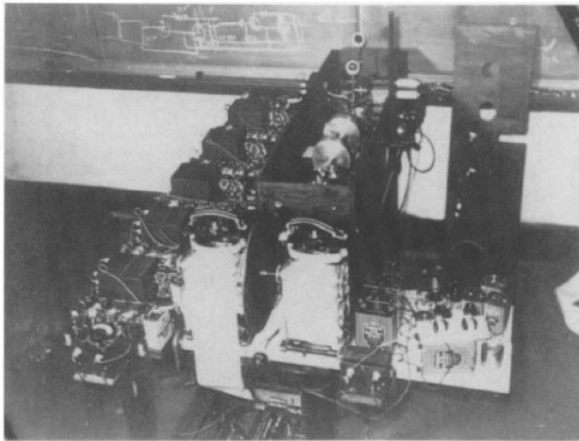


Fig. 2 IL predittore antiaereo, Tratta da Wiener da D. I. C. 5980 A. A. Directors, «Summary Report for Demonstration». In Galison, 1994, p.54 trad. it.

b. Questione di feedback

Stibitz annotava nel suo diario - lo stesso giorno in cui aveva sperimentato con Weaver la sensazione perturbante alla vista del predittore di Wiener, come la macchina rappresentasse la più grande ambizione di questi, cioè dare ragione del comportamento: “W[iener] fa notare che la loro apparecchiatura è probabilmente uno degli approcci meccanici più vicini mai compiuti al comportamento fisiologico” [Stibitz, 1942 D, in Galison, 1994, p.57-58 trad. it.].

Come notato, Wiener fu instancabile nell’universalizzare le conseguenze del suo lavoro sul predittore. Il rapporto tra comportamento e retroazione fu generalizzato nel fondamentale articolo *Behaviour, Purpose and Teleology* del 1943 [Rosenblueth et al, 1943] scritto da Wiener, Bigelow e dal cardiologo messicano, ai tempi in visita ad Harvard, Arturo Rosenblueth [1900-1970]³⁶. Questo saggio di poche pagine pagine - un testo fondativo delle nuove scienze

³⁶ L’importanza della figura di Rosenblueth nella vita e sviluppo del programma di ricerca di Wiener trova testimonianza nel fatto che *Cibernetica* è dedicata proprio allo scienziato messicano. “Ad Arturo Rosenblueth per molti anni mio compagno di scienza “. Anche le primissime parole dell’*Introduzione* alla prima edizione del 1948, sono dedicate all’incontro con Rosenblueth. Racconta Wiener di aver incontrato Rosenblueth durante una delle riunioni mensili dedicate a discussioni sul metodo scientifico organizzate da questi durante la permanenza presso

cibernetiche [Cfr. Pasquinelli, 2023, p.139] - si proponeva due scopi: “Il primo è definire lo studio comportamentistico di eventi naturali e classificare il comportamento, il secondo mettere in risalto il concetto di fine” [Rosenblueth et al., 1943, p.91 trad. it].

Lungi da essere una semplice analisi classificatoria delle tipologie di comportamento, l'articolo, secondo gli autori, “rivela che un'analisi uniforme del comportamento è applicabile alle macchine come agli organismi viventi” [Rosenblueth et al., 1943, p.100] e come questa sia possibile mediante la corretta considerazione del concetto di retroazione, di feedback.

Dopo aver delineato lo studio comportamentista di un oggetto come l'esame dell'uscita (output) e relazioni di questa con l'entrata (input) - intendendo con output i cambiamenti di qualsiasi natura prodotti dall'oggetto sull'ambiente e con input ogni evento esterno che lo modifica in qualche maniera - Wiener, Bigelow e Rosenblueth definiscono il comportamento come “ogni cambiamento di un'entità rispetto al suo ambiente” [Rosenblueth et al., 1943, p.92 trad. it].

Questa definizione iniziale, che collima con quella data precedentemente di servomeccanismo, è talmente ampia da essere priva di ogni utilità se non fosse ulteriormente approfondita. Il primo passo è la definizione di comportamento attivo, e cioè “quello in cui l'oggetto è la fonte dell'energia di uscita implicata da una specifica reazione” [Rosenblueth et al., 1943, p.92 trad. it].

A sua volta il comportamento attivo può essere diviso in due classi: *casuale* o *volto a un fine*. Il primo è il comportamento che può essere interpretato come quello che non mira a uno scopo, mentre quello rivolto a un fine indica che “il comportamento può essere interpretato

la Harvard Medical School e tenute nella Vanderbilt Hall dell'ateneo statunitense. L'incontro avvenne quando Wiener fu invitato a partecipare a queste riunioni dal suo ex allievo, il dottor Manuel Sandoval Vallarta. Wiener da sempre appassionato di questioni di filosofia, epistemologia e di metodologia delle scienze ricorda l'impatto duraturo nella sua formazione di quegli incontri aperti anche a non-medici: “Si conversava animatamente e liberamente. Non era quello un luogo ove ci si potesse sentire incoraggiati a mantenere il proprio riserbo. Dopo il pranzo, qualcuno del nostro gruppo o un invitato leggeva una comunicazione su qualche argomento scientifico, in genere argomenti in cui erano i problemi metodologici ad avere un'importanza primaria, o per lo meno considerevole. L'oratore doveva quindi affrontare una critica penetrante, cordiale ma inesorabile. Era una perfetta catarsi per le idee non ben maturate, per l'insufficiente autocritica, per l'eccessiva presunzione e per l'ampollosità. Chi non riusciva a sopportare la cosa non tornava più, ma tra i vecchi frequentatori abituali di quelle riunioni ve n'è più d'uno convinto che esse diedero un contributo importante e duraturo alla nostra formazione scientifica” [Wiener, 1948, pp. 23-24]. Ad unire intellettualmente Wiener e Rosenblueth vi era anche la comune convinzione che, data la crescente specializzazione delle scienze, che aveva fatto tramontare l'epoca di grandi geni che avevano il pieno dominio di tutti gli ambiti del sapere come Leibniz, “santo patrono della cibernetica» [Ivi, p.35], «le aree più propizie allo sviluppo delle scienze fossero quelle che erano state trascurate come terra di nessuno fra i vari campi esplorati” [Ivi, p. 24]. Il carattere interdisciplinare della cibernetica si deve anche alla duratura influenza del rapporto tra i due studiosi. Rosenblueth fu protagonista centrale degli incontri interdisciplinari tra scienziati che scaturirono dalle conseguenze del suo lavoro con Wiener

come diretto al raggiungimento di un obiettivo, cioè di una condizione finale in cui l'oggetto di cui si studia il comportamento raggiunge una correlazione definita nello spazio o nel tempo rispetto a un altro oggetto o evento" [Rosenblueth et al., 1943, pp.92-93 trad. it.]. La tassonomia delle azioni proposta da Wiener, Rosenblueth e Bigelow non si interrompe a questa suddivisione, ma introduce una distinzione cruciale: lo stesso comportamento attivo rivolto a un fine può essere suddiviso in due classi: "con retroazione [feedback] (o "teleologico") e "senza retroazione" [non feedback] (o "non teleologico")" [Rosenblueth et al., 1943, pp.94-95 trad. it.]. Lo stesso feedback può essere di due tipi: *positivo*, quella in cui "la retroazione accresce i segnali d'ingresso ma non li corregge" e *negativo*: quello in cui vi è una correzione rispetto al margine d'errore in cui l'oggetto si trova in un determinato momento rispetto all'obiettivo, cioè "i segnali dall'obiettivo sono usati per limitare le uscite, che altrimenti supererebbero l'obiettivo" [Rosenblueth et al., 1943, p.95 trad. it.]. È questo secondo tipo di feedback, quello in cui la retroazione si oppone a quanto il sistema sta facendo per correggerlo, quello che interessa agli autori. «Ogni comportamento rivolto a un fine può essere considerato un processo che richiede una retroazione negativa» [Rosenblueth et al., 1943, p.95 trad. it.] e questo è valido sia per gli organismi viventi e sia per le macchine.

"Là dove Darwin ha assiduamente seguito le tracce delle similarità tra essere umano e animale allo scopo di rendere indistinto il confine tra essi, i tentativi di Wiener erano dedicati a cancellare la distinzione tra essere umano e macchina. Il cane di Darwin soffriva di rimorso; il predittore antiaereo di Wiener aveva la preveggenza" [Galison, 1994, trad. it. p. 60].

Il comportamento finalizzato a feedback negativo può essere sottoposto a una ulteriore divisione, tra comportamento estrapolante e non-estrapolante, basata sulla capacità di predizione (con quest'ultima che può essere a sua volta suddivisa in primo, secondo, terzo ordine e così via), esattamente come la macchina che Wiener e Bigelow avevano realizzato [Rosenblueth et al., 1943, p.97 trad. it.].

La sistematizzazione [fig. 3] delle tipologie di comportamento proposta da Wiener, Bigelow e Rosenblueth non è frutto di un mero piacere classificatorio, ma mette in evidenza alcune caratteristiche rilevanti. "Suggerisce la possibilità di progettare test sistematici sempre più complessi del comportamento degli organismi" [Rosenblueth et al., 1943, p.100 trad. it.]. Essa pone inoltre "in rilievo i concetti di fine e di teleologia, concetti che, anche se oggi sono piuttosto screditati, hanno una grande importanza" [Rosenblueth et al., 1943, p.100 trad. it.] e rivela come, dal punto di vista del comportamento regolato da feedback, la differenza tra uomo/organico e macchina/artificiale tenda a sfumare. "Finalmente essa rivela che un'analisi

uniforme del comportamento è applicabile alle macchine come agli organismi viventi, a prescindere dalla complessità del comportamento” [Rosenblueth et al., 1943, p.100 trad. it.]. Il metodo di studio per analizzare entrambi i gruppi, infatti, non cambia. Il sistema a retroazione negativa diviene, dal punto di vista dell’azione finalizzata³⁷, la base per lo studio di qualunque sistema.

Che il sistema da studiare sia artificiale o biologico, il metodo non muta: “come oggetti di ricerca scientifica, *gli esseri umani non differiscono dalle macchine*” [Wiener, Rosenblueth, 1950 enfasi mia]³⁸.

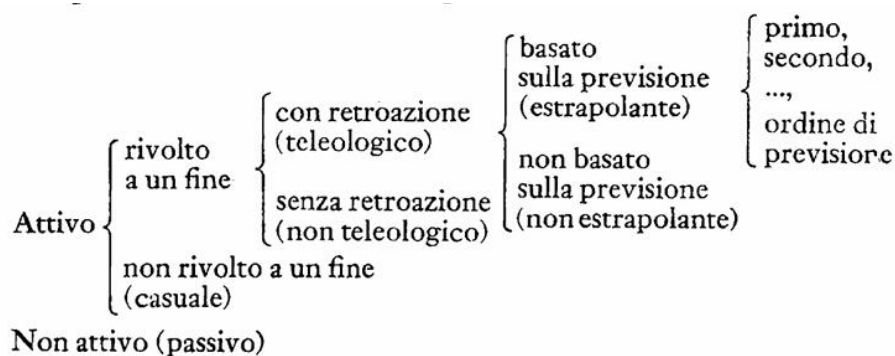


Fig. 3. La classificazione del comportamento in *Behavior, Purpose and Teleology* di Rosenblueth, Wiener e Bigelow, 1943, p.99 trad. it.

³⁷ Lo studio *funzionale*, secondo Wiener, Bigelow e Rosenblueth, si oppone all’analisi comportamentista in senso strettamente *behaviouristico*, perché «l’oggetto principale è l’organizzazione intrinseca dell’entità studiata, la sua struttura e le sue proprietà». «Mentre l’analisi comportamentistica delle macchine e degli organismi viventi è in gran parte dello stesso tipo, il loro studio funzionale rivela differenze profonde». Gli organismi viventi, per i tre autori, sono composti di materiale colloidale prevalentemente proteico e si distinguono dalle molecole semplici che costituiscono il materiale metallico delle macchine. Tuttavia, a mio avviso, anche questa differenza strutturale tra sistemi organici e meccanici tende a sfumare in favore di un’analisi indifferenziata. È solo lo stato dell’arte tecnologico a mantenere viva la distinzione sul piano funzionale, mentre quella comportamentale scompare con l’accento posto sui sistemi a feedback negativo. Rosenblueth, Wiener e Bigelow affermano che, se oggi dovessimo costruire un robot, useremmo metallo e valvole e non colloidali proteici. Ma ciò dipende soltanto dal livello di avanzamento scientifico. “In futuro, quando la nostra conoscenza dei colloidali e delle proteine sarà maggiore, gli ingegneri potranno provare a progettare dei robot con il comportamento e in più anche la struttura simile a quella di un mammifero. Il modello insuperabile di un gatto è ovviamente un altro gatto, sia che sia nato da un gatto o che sia stato ottenuto per sintesi in laboratorio” [Rosenblueth et al., 1943, pp.102-103].

³⁸ “As objects of scientific enquiry, humans do not differ from machines”.

Questa è una posizione fondamentale. Dal punto di vista scientifico non vi è distinzione tra uomo e macchine, entrambi i sistemi, individuata la caratteristica fondamentale nel feedback negativo, per essere studiati, vanno ricondotti a una base comune.

La nascente scienza cibernetica, la generalizzazione del modello del predittore, produce conseguenze psicoogiche, ontologiche e antropologiche notevolissime:

Wiener giunse a vedere il predittore come un prototipo non soltanto della mente di un inaccessibile pilota dell'Asse avversario ma anche dell'artigliere antiaereo Alleato, e poi da un punto di vista ancor più allargato si spinse fino a includere il vasto spettro dei sistemi umani di retroazione propriocettiva ed elettrofisiologica. [Galison, 1994, trad. it. p. 41].

Il concetto di retroazione negativa, elaborato da Wiener, Rosenblueth e Bigelow come principio regolatore del comportamento e base comune dei sistemi biologici e artificiali, è strettamente connesso all'idea di regolazione intesa come tendenza all'equilibrio tra sistema e ambiente. Questo principio richiama un'idea che sarebbe progressivamente divenuta, come osservano Conway e Siegelman, un "principio [che] sarebbe divenuto un dogma centrale della nuova scienza di Wiener" [Conway, Siegelman, 2005, p.22 trad. it.], con ricadute anche di carattere sociologico, estese a tutto il movimento cibernetico: il concetto di *omeostasi*.

Le radici di questo principio, come riconosce lo stesso Wiener [Wiener, 1951], si trovano nelle scienze del vivente. Esse risalgono in particolare agli studi del fisiologo francese Claude Bernard (1813-1878), che nei suoi lavori sulla digestione - *Du suc pancréatique et de son rôle dans les phénomènes de la digestion* [Bernard, 1849; cfr. Cooper, 2008] - aveva già intuito l'esistenza di meccanismi regolatori dell'equilibrio interno dell'organismo.

A sviluppare ulteriormente questo principio di regolazione tendente all'equilibrio fu uno degli scienziati più influenti nello sviluppo intellettuale di Wiener: Walter Cannon (1871-1945), fisiologo di Harvard, amico del padre di Wiener Leo, nei cui laboratori il Norbert bambino restava affascinato dai "meravigliosi, innovativi congegni scientifici" [Conway, Siegelman, 2005, p.22 trad. it.] che vi si trovavano.

Cannon coniò il termine "*omeostasi*" [*homeostasis*] nel 1929 in *Organization for physiological homeostasis* [Cannon, 1929], per descrivere ciò che Bernard aveva anticipato: "il meccanismo organico del corpo atto a mantenere uno stato di equilibrio interno sano e stabile attraverso un bilanciamento di azioni e reazioni a regolazione automatica: sudare per raffreddare il corpo quando si surriscalda, rilasciare ormoni per eccitare o inibire l'azione degli organi e delle cellule nervose" [Conway, Siegelman 2005, p.22 trad. it.]

Wiener, in *Cibernetica*, sottolinea il legame tra il concetto di omeostasi e la retroazione negativa, osservando come “un vasto insieme di casi in cui qualche tipo di feedback non solo è individuabile nei fenomeni fisiologici ma è assolutamente essenziale alla continuazione della vita rientra in ciò che è noto come omeostasi” [Wiener, 1948, p. 156, trad. it.]. Egli nota inoltre che, se fino a pochi anni prima dei suoi studi – con poche eccezioni come i meccanismi di regolazione delle macchine a vapore analizzati da Clerk Maxwell – “sarebbe stato difficile trovare al di fuori del corpo umano dei meccanismi analoghi a quelli omeostatici”, al tempo in cui scrive sistemi “retroattivi (o a feedback)” erano ormai largamente diffusi [Wiener, 1951, p.117 trad. it.].

In questi sistemi “un errore di funzionamento è reintrodotta nel sistema e contribuisce a determinare il suo funzionamento futuro”, così che “le oscillazioni nel funzionamento vengono corrette dall’inizio, senza aspettare che assumano grandi proporzioni”. La “valvola termoionica,” (futuro analogo funzionale del neurone) osserva Wiener, era divenuta “uno dei mezzi più economici per introdurre queste correzioni e per effettuare le combinazioni che le rendono possibili” [Wiener, 1951, p. 118, trad. it.].

Quasi a simbolo del concetto stesso di omeostasi, emerge un altro “mostro cibernetico” [Pickering, 2005]: l’*omeostato* [*Homeostat*] inventato e realizzato, per la prima volta nel 1948, dallo psichiatra e cibernetico britannico William Ross Ashby (1903–1972). Questa macchina, prodotta in diverse versioni, era costruita come “una specie di servomeccanismo autostabilizzantesi” [Shannon, 1953, p.104 trad. it.]. L’omeostato era formato da quattro unità identiche, ciascuna con un magnete montato su un perno e collegato a un circuito elettrico che ne registrava e trasmetteva le deviazioni dalla posizione centrale. Le correnti generate da tali deviazioni venivano inviate alle altre unità, creando un sistema di interazioni reciproche: i movimenti di un magnete influenzavano le correnti che, a loro volta, modificavano i movimenti degli altri. Ogni unità era dotata di vaschette d’acqua ed elettrodi che fornivano un gradiente di potenziale elettrico, trasformando le variazioni meccaniche in segnali elettrici, mentre potenziometri e commutatori regolavano intensità e direzione delle correnti. L’insieme appariva come un dispositivo capace di reagire continuamente a perturbazioni interne ed esterne, mantenendo attivo un complesso circuito di retroazioni incrociate [Cfr. Ashby, 1960, pp. 141–148, trad. it.].

Ashby definiva “*stabile*” un sistema “in grado di correggere o regolare il proprio comportamento sulla base di un semplice dispositivo di retroazione” [Cordeschi, 1998, p. 42]. L’omeostato, tuttavia, rappresentava l’implementazione di un sistema “*ultrastabile*” [Cfr. Ashby,

1960, pp. 115–132, trad. it.], cioè un sistema capace di modificare la propria organizzazione interna in funzione dell'ambiente, attraverso un meccanismo di feedback, tentativi ed errori e "raccolta delle informazioni" [Ashby, 1960, p. 119]. Ashby sintetizzava l'*ultrastabilità* in questo modo: "se il tentativo fallisce, cambia il modo di comportarti; quando e solo quando il tentativo riesce, mantieni questo modo di comportarti" [Ashby, 1960, p. 121, trad. it.].

"L'omeostato, con tutti i suoi limiti, indicava alla modellistica cibernetica il giusto livello di astrazione per il modello, essendo l'*ultrastabilità*, nelle sue caratteristiche essenziali, il principio che Ashby riteneva comune all'organizzazione del cervello e a quella della macchina" [Cordeschi, 1998, p. 211]: un principio che accomunava vivente e non vivente, biologico e artificiale, macchina e cervello.

Ashby espose le sue idee alla platea interdisciplinare degli scienziati che si riunivano nel movimento cibernetico delle *Macy Conferences* del 1952:

Considererò dunque l'organismo come un meccanismo che affronta un mondo ostile e difficile e che ha come compito fondamentale quello di mantenersi in vita. [...] Ciò, naturalmente, non è in alcun modo peculiare degli organismi viventi. Un ingegnere, seduto al pannello di controllo di una nave, ha esattamente lo stesso compito da svolgere. Davanti a sé ha una serie di quadranti, alcuni dei quali rappresentano variabili essenziali della nave, ed è suo compito condurre le operazioni in modo che le lancette restino sempre entro i limiti corretti. *Il problema, dunque, è uniforme tra l'inanimato e l'animato* [Ashby, 1952, in Pias, 2016 p. 593, enfasi mia].

Alla base del mantenimento dell'omeostasi si trova la retroazione negativa: "La retroazione negativa fu anche il meccanismo adottato per far funzionare l'omeostasi. I segnali di feedback possono essere utilizzati per mantenere la stabilità contro le fluttuazioni imposte dall'esterno. Inoltre, la retroazione negativa può essere impiegata per 'convergere' verso uno stato finale - cioè per avvicinarsi a una condizione stabile e, una volta raggiunta, mantenerla" [Cooper, 2008, trad. mia]³⁹.

Le considerazioni cibernetiche sull'omeostasi – che si estenderanno, come vedremo, fino a comprendere ogni sistema, compreso quello sociale e i sistemi biologici e artificiali che lo abitano – offrono anche una chiave per intendere scientificamente il comportamento come

³⁹ "Negative feedback was also the mechanism adopted to make homeostasis work. Feedback signals can be used to maintain stability against externally imposed fluctuations. In addition, negative feedback can be used to 'home in' on an end state-that is, to approach a stable condition and, once achieved, to maintain it. Wiener sub-titled his influential book on cybernetics, "control and communication in the animal and the machine". Si rimanda allo stesso [Cooper, 2008] per l'emergere storico e l'analisi del concetto di *omeostasi*.

regolato dal feedback. È importante sottolineare che la cibernetica classica scelse deliberatamente di descrivere l'intenzionalità unicamente attraverso la relazione tra comportamento osservabile e ambiente esterno. Gli elementi soggettivi – considerati per definizione “vaghi” [Ashby, 1940] – vennero esplicitamente esclusi, così da restringere e rendere più precisa la definizione operativa di intenzionalità proposta da Rosenblueth, Wiener e Bigelow nel 1943. Come avrebbe poi ribadito Ashby, la cibernetica, “in quanto scienza che ha interamente a che fare con ciò che è osservabile e dimostrabile”, “non ha niente da dire” sugli aspetti soggettivi o interni del comportamento umano [Cordeschi, 1994, p. 24].

In quest'ottica si può leggere una lettera che Ashby inviò a Wiener, commentando il valore del lavoro esemplificato in *Behaviour, Purpose and Teleology*:

Quando considero come gli psicologi hanno cercato di risolvere esattamente questo problema per decenni (se non secoli), la scatola nera essendo il cervello, e quando penso a quanta poca attenzione hanno dato ai principi implicati, la mia opinione degli psicologi cade sempre più in basso. Il guaio con lo psicologo è che è troppo orgoglioso per imparare a camminare prima di provare a correre. Così oggi sta disteso sul suo posteriore, scioccamente dondolando le sue gambe e pretendendo di essere un ballerino, quando in realtà non ha ancora imparato ad andare carponi. Per questa ragione lo considero un grande complimento il dire che il tuo studio del problema della “scatola nera” è un primo passo verso una psicologia scientifica! [Ashby, lettera a Wiener, 6 febbraio 1951, in Galison, 1994, pp. 66–67, trad. it.].

A queste considerazioni sulla psicologia del comportamento si può sommare un ulteriore sviluppo nella comprensione della natura umana che lo studio del nemico come elemento di imprevedibilità computazionale ha introdotto. Non ci troviamo più di fronte a un'immagine disumanizzata, che degrada il nemico all'animalità e lo dipinge come irrazionale, sub-umano e bestiale – come accadeva, ad esempio, con la rappresentazione del nemico giapponese da parte dei soldati Alleati, in particolare quelli dei paesi anglofoni. Né si tratta di una visione neutrale, in cui gli obiettivi umani del nemico sono freddamente considerati, grazie al distacco dovuto alle migliaia di metri di altitudine da cui venivano sganciate le bombe, alla stregua di normali bersagli da distruggere [Cfr. Galison, 1994, p. 42-44 trad. it].

L'immagine dell'umano che scaturiva dal predittore – e che avrebbe alimentato la cibernetica e la visione del mondo da essa ispirata – lasciò forse un'impronta più duratura. Il nemico veniva concepito come fuso con gli strati di acciaio della macchina che guidava e che occorreva abbattere: un'entità in cui la distinzione tra artificiale e naturale perdeva significato.

Allo stesso modo, l'artigliere risultava ormai indistinguibile dall'arma stessa, "un puntatore umano o un capopezzo, o entrambi, collegati al sistema di puntamento e funzionanti come parte essenziale di questo. È indispensabile conoscere le loro caratteristiche, per incorporarle matematicamente nelle macchine che essi controllano" [Wiener, 1948, p. 29, trad. it.].

Di entrambi gli elementi umani, ormai fusi con le macchine che manovravano e resi indistinguibili da esse, ciò che contava era il comportamento, la cui componente essenziale era la retroazione, il feedback negativo. Questo dissolversi del confine tra umano e meccanico condusse Wiener a concepire i diversi sistemi – incluso l'uomo – come *black box*, scatole nere. "Scatole nere, per come Wiener utilizzò il termine, stava a significare un'unità deputata a svolgere una funzione senza che se ne conoscesse il funzionamento interno; *scatole bianche* indicava invece la possibilità di specificarne i meccanismi" [Galison, 1994, p. 61, trad. it.].

La spiegazione a *black box* generava⁴⁰, *a priori*, spiegazioni del comportamento che erano sfuggite al comportamentismo classico [Cfr Galison, 1994 p.56 trad. it.]. La proibizione, sancita dai comportamentisti prima del secondo conflitto mondiale, di occuparsi di concetti ritenuti pseudoscientifici come gli "stati mentali" [Watson, 1920, trad. it., p. 154] e l'abbandono dell'introspezione, sostituita dall'imperativo secondo cui il metodo della psicologia, in quanto branca delle scienze naturali, "è la previsione e il controllo del comportamento" [Watson, 1913, trad. it., p. 53], trovano negli studi di Wiener un ulteriore e radicale superamento.

Espulso "lo spettro dalla macchina" [Ryle 1949, p. 11 trad it] le intenzioni di un nemico sepolto sotto strati e strati di acciaio, se non fuso con esso, sono identificate, quasi in maniera automatica, con i meccanismi di retroazione.

⁴⁰ Il problema della *black box* nella modellazione scientifica – centrale nella ricerca su mente e calcolatori – fu affrontato da Wiener e Rosenblueth nel 1945 in relazione al ruolo dei modelli nella spiegazione scientifica, in *The Role of Models in Science*. In questo saggio i due autori analizzano il valore dei modelli, considerati imprescindibili per lo studio e la predizione dei sistemi complessi. La modellizzazione implica necessariamente, secondo loro, l'utilizzo di spiegazioni a "scatola chiusa" [*black box*], in cui "un certo numero finito di variabili in entrata determina un certo numero finito di variabili in uscita". È il grado di conoscenza della relazione tra queste variabili a consentire l'ampliamento della comprensione di un sistema e l'"apertura" delle *black box*: altrimenti, sistemi che presentano la stessa relazione fra variabili di input e output sarebbero strutturalmente indistinguibili. La modellizzazione può crescere asintoticamente ma, se portata al limite, diventa impraticabile, perché il modello finirebbe per coincidere con il sistema stesso, rendendosi inutile. "Al limite [il modello] diventerà proprio quel sistema. Come a dire, per fare un esempio specifico, che il miglior modello materiale di un gatto è un altro gatto: o anzi, meglio ancora, proprio quello stesso gatto" [Rosenblueth, Wiener, 1945, pp.86-82 trad. it.]. Lewis Carroll aveva espresso questo concetto in un episodio di *Sylvie e Bruno*, mostrando come l'unica mappa di un paese soddisfacente in tutti i dettagli non possa che coincidere con il paese stesso. Analogamente, scrivono i due autori, "Il modello formale ideale sarebbe quello che è in grado di coprire l'universo intero, cui si adeguerebbe per complessità attraverso una corrispondenza biunivoca. Chiunque avesse la capacità di elaborare e comprendere nella sua interezza un modello siffatto, finirebbe per trovarlo superfluo, potendo egli cogliere l'universo intero come un tutto. Costui possiederebbe la terza categoria di conoscenza descritta da Spinoza" [Rosenblueth, Wiener, 1945, pp.86-82 trad. it.].

Possiamo adesso rispondere alla domanda sul perché Stibitz e Weaver furono perturbati dai risultati del predittore di Wiener. Il perturbante, l'*unheimlich* è «quella sorta di spaventoso che risale a quanto ci è noto da lungo tempo, a ciò che ci è familiare» [Freud, 1919, p.270 trad. it], Secondo Freud, citando Jentsch, una “condizione particolarmente favorevole al sorgere di sentimenti perturbanti si verifica quando predomina l'incertezza intellettuale se qualcosa sia o no vivente, o quando ciò che è inanimato spinge troppo oltre l'analogia con ciò che è vivo” [Ivi, p.185].

Il perturbante sorgerebbe quando viene meno la sicurezza intorno alla natura inanimata di un oggetto. Il manifestarsi di caratteristiche proprie ad entità viventi, animate, umane, sarebbe l'innescò di questo spiacevole stato d'animo. Questo significa, se tradotto nel linguaggio delle scienze cognitive successive alle riflessioni dello psicanalista viennese, come nota Peter Galison, a proposito della riflessione di S. Covell sul tema, che “il perturbante riflette precisamente l'ansietà filosofica esacerbata dall'ambiguità creata quando non è chiaro se è a portata di mano una mente o un mero oggetto inanimato” [Galison, 1994, p.57 n.38 trad it]. Il problema del perturbante deriva dall'inveterato problema filosofico del solipsismo e dell'esistenza di altre menti.

Se l'antropomorfismo è “un processo cognitivo automatico di attribuzione della natura umana o di caratteristiche uniche dell'essere umano a oggetti ed entità non umane, tecnologiche, spirituali o naturali, che provoca una miscela di risposte affettive, comportamentali e cognitive” [Ruijten 2015, in Roselli et al., 2025, trad. mia]⁴¹, il predittore di

⁴¹ “In line with this latter definition, Ruijten (2015) described anthropomorphism as an automatic cognitive process of attributing human nature or human uniqueness characteristics to nonhuman technological, mental, or natural objects, which elicits a mixture of affective, behavioral, and cognitive responses”. Per una tassonomia e l'analisi delle varie tipologie di antropomorfismo in relazione ai contesti culturali si veda Roselli, et al, 2025. Un classico intorno alle reazioni *Uncanny* di fronte a robot e entità (più o meno) antropomorfe è Mori, 1970. Le considerazioni qui fatte richiamano al problema dell'attribuzione dell'intenzionalità ad agenti artificiali al fine di prevederne il comportamento.. Su questo tema si rimanda a Dennett, 1987 e alla sui studi sull'*intentional stance*: “Ecco come funziona: per prima cosa si decide di trattare l'oggetto, il cui comportamento deve essere previsto, come un agente razionale; poi si determinano le credenze che tale agente dovrebbe avere, dato il suo posto nel mondo e il suo scopo. Successivamente si stabiliscono i desideri che dovrebbe avere, sulla base delle stesse considerazioni, e infine si predice che questo agente razionale agirà per perseguire i propri obiettivi alla luce delle sue credenze. Un po' di ragionamento pratico a partire dall'insieme scelto di credenze e desideri condurrà, nella maggior parte dei casi, a una decisione su ciò che l'agente dovrebbe fare; ed è proprio questo ciò che si prevede che l'agente farà” [Here is how it works: first you decide to treat the object whose behavior is to be predicted as a rational agent; then you figure out what beliefs that agent ought to have, given its place in the world and its purpose. Then you figure out what desires it ought to have, on the same considerations, and finally you predict that this rational agent will act to further its goals in the light of its beliefs. A little practical reasoning from the chosen set of beliefs and desires will in most instances yield a decision about what the agent ought to do; that is what you predict the agent will do, Dennett, 1987, p.17 trad. mia]

anthropomorphism as an automatic cognitive process of attributing human nature or human uniqueness characteristics to nonhuman technological, mental, or natural objects, which elicits a mixture of affective, behavioral, and cognitive responses”.

Wiener non suggeriva affatto forme umanoidi o somiglianze esteriori con l'uomo - in quello che è possibile definire "antropomorfismo oggettivo" [Cfr. Roselli et al., 2025] Eppure Stibitz e Weaver, turbati dal suo comportamento, gli attribuirono caratteristiche umane: un caso di "antropomorfismo interpretativo", ossia l'assegnazione di "intenzioni, credenze ed emozioni ad agenti non umani [...] sulla base del loro comportamento" [Roselli et al., 2025, trad. mia]⁴².

Il predittore⁴³ antiaereo di Wiener era appunto un oggetto inanimato, meccanico, un insieme antiestetico di metallo, valvole e circuiti, ma possedeva la caratteristica propria di ogni essere vivente, e dell'essere umano, il *feedback*, un comportamento basato su un sistema di retroazione negativa.

A turbare i due superiori di Wiener non fu dunque l'aspetto della macchina, bensì la funzione che essa realizzava: la capacità di prevedere e abbattere un velivolo nemico, proprio ciò che in quel contesto storico e culturale veniva percepito come il tratto distintivo dell'umano. In questo senso, il predittore - e con esso la cibernetica che lasciava intravedere - apriva a un'idea nuova: "quella di una teoria di cui gli organismi e le macchine, data la comune organizzazione funzionale dei rispettivi sistemi di azioni, fossero per certi aspetti realizzazioni diverse" [Cordeschi, 1998, p. 43].

La concezione di ciò che costituisce la caratteristica essenziale di un essere animato varia storicamente, così come la concezione di ciò che può essere considerato mente. Ed è proprio in riferimento al contesto bellico - alla lotta contro un nemico dotato di apparecchiature letali, su un campo di battaglia sempre più impersonale e meccanizzato - che persino quella che può apparire una palese iper-semplificazione del *τί ἐστι* di un organismo animato, e addirittura dell'essere umano stesso, diventa automatica⁴⁴.

Stibitz e Weaver rimasero perturbati perché nella macchina di Wiener avevano scorto le caratteristiche dell'uomo così come ridefinite dalla drammatica situazione storica che stavano vivendo: una condizione segnata dal dominio della macchina, dal continuo alternarsi di mosse

⁴² "interpretative anthropomorphism, i.e., attributing intentions, beliefs, and emotions to nonhuman agents (for example, animals) based on their behavior."

⁴³ Data l'importanza non solo tecnica, ma addirittura fisiologica, psicologica e filosofica assunto dal concetto di predizione le parole di Bigelow a riguardo possono essere significative. "È necessario fare assoluta chiarezza sul concetto di previsione". "In senso stretto e assoluto, nessun operatore di rete, o operatore umano, è in grado di prevedere il futuro di una funzione di tempo". "I cosiddetti *anticipi*, valutati attraverso le reti o qualsiasi altro strumento, sono in realtà ritardi (funzioni del passato conosciuto), artificialmente rovesciati e aggiunti al valore presente della funzione" [Dyson, 2012, p.131 trad. it.]

⁴⁴Per le critiche rivolte a Wiener circa la riduzione dell'essere umano - dei suoi scopi, moventi e obiettivi - a un mero sistema di retroazione, esemplificate da quelle mosse da Richard Taylor (1919- 2003), filosofo della Brown University [Taylor, 1950] si veda Galison, 1994, pp. 63-65.

e contromosse, contraddistinte da una razionalità fredda e implacabile; nemici resi indistinguibili da strati su strati di metallo e alleati la cui azione si era ormai fusa con l'arma che utilizzavano.

Il contesto bellico aveva talmente normalizzato l'associazione tra le caratteristiche dell'azione umana e le attività delle macchine che era divenuto abituale, tra gli aviatori, riferirsi ai propri velivoli - dotati di numerosi servomeccanismi a retroazione - in termini quasi animistici, parlando scherzosamente di "spiritelli" [*gremlins*] che abitavano le macchine [Wiener, 1945, in Galison, p. 60, trad. it.].

c. Il seme cibernetico nell'era dell'informazione

Il lavoro di Wiener, Bigelow e dei loro due collaboratori, iniziato con uno stanziamento da parte dell'NDRC di un totale di soli 2325 dollari, non ha esaurito la sua funzione con la sola produzione di un macchinario dalla dubbia utilità pratica, ma è stato gravido di conseguenze che, nate nella tempeste della Seconda Guerra Mondiale, segnano la forma, tecnica e culturale che hanno assunto le tecnologie digitali e le ICT che plasmano la contemporaneità.

Intorno a queste conseguenze si possono fare delle considerazioni a partire dal punto di vista della metodologia matematica, che sarà analizzata maggiormente nello specifico più avanti in questo capitolo, individuata per essere il sostrato algoritmico del predittore. Si prenda in esame l'apparecchio predisposto per essere l'analogo del comportamento del sistema artigliere-cannone-pilota-velivolo e costruito per generare traiettorie randomizzate simili al comportamento umano con cui programmare la macchina per le predizioni secondo il metodo statistico, che, come ricordato, rappresentava la novità della proposta di Wiener «per vincere la guerra con una mano sola» [Dyson, 2012, p.126 trad.it.]. Dal lavoro di ingegneria propriocettiva messo appunto da Wiener e Bigelow emersero, in particolare dalle "reazioni nervose" dello pseudo pilota, due risultati: "primo, non c'era alcuna particolare correlazione tra le fluttuazioni registrate di operatori differenti; secondo, c'era un alto grado di autocorrelazione tra le prime prestazioni dell'operatore e le seguenti" [Galison, 1994, p.59 trad. it.]. Le prestazioni dello stesso pilota in condizioni simili erano coerenti tra di loro, il che significava che il metodo statistico di predizione dipendeva dalla storia delle performance del pilota. In questa dimensione riferito alla tendenza di schemi del passato a ripresentarsi in futuro.

Lo confermano le annotazioni sul diario di lavoro di Stibitz riportate nella stessa giornata del 1° luglio: "Tra parentesi, il predittore di Wiener è basato su buone idee

comportamentiste, poiché tenta di predire le azioni future di un organismo non studiando la struttura dell'organismo ma studiando il comportamento passato dell'organismo» [Stibitz, 1942 D, in Galison, 1994, p.57-58 trad. it.]

Il predittore allora doveva essere alimentato, per essere efficace, con dei dati reali riguardanti altrettanto reali prestazioni di volo. Durante l'estate del 1942 Wiener e Bigelow fecero il giro di varie installazioni militari incaricate di misurare accuratamente le prestazioni di volo degli aerei⁴⁵. Dopo aver ottenuto dati sulle regolarità e le irregolarità del moto degli aeroplani, indicazioni metodologiche sugli errori di procedure di tracciamento, alla fine ottennero le informazioni più preziose dall'Anti-Aircraft Board a Camp Davis, North Carolina. Queste erano «i dati di rotta di due voli di prova – i cosiddetti voli 303 e 304 – a intervalli di un secondo» [Galison, 1994, p.58] Questo data set era fondamentale, perché rappresentava realisticamente delle traiettorie realistiche utilizzabili come segnali di ingresso nel predittore e come test sul suo *output*.

La macchina antiaerea di Wiener, fondata sul metodo statistico e sull'idea di servomeccanismo a retroazione negativa, altro non era che una “macchina ad apprendimento”, una *learning machine* [Wiener, 1948, p.219 trad. it.]. Quasi vent'anni dopo la costruzione e le prime prestazioni del predittore, quest'ultimo macchinario rimase per Wiener l'archetipo di macchina cibernetica capace di apprendere. Nella prefazione alla seconda edizione di *Cibernetica* del 1961, le caratteristiche del metodo statistico per prevedere le traiettorie dei velivoli avversari sono poste in stretta relazione con la capacità della macchina di apprendere dall'esperienza.

⁴⁵ Wiener e Bigelow compirono più di un viaggio dal MIT a sud, in Virginia e Carolina del Nord, presso le basi aeree militari [Wiener, 1953b, p.249]. Lo stesso Wiener lo dichiara nel suo report del dicembre 1942: “Ho visto le risposte di un buon numero di questi ufficiali dell'esercito e, inoltre, ho avuto l'opportunità di effettuare uno studio statistico approfondito sui voli 303 e 304, fornito dal Consiglio Anti-Aereo presso Camp Davis, North Carolina” [“I have seen the answers of a good many of these army officers, and I have in addition had the opportunity to make a thorough statistical study of flights 303 and 304 furnished me by the Anti Aircraft Board at Camp Davis, North Carolina”, Wiener, 1942, trad. it. mia]. Conway e Siegelman nella loro biografia di Wiener notano come questi viaggi furono istruttivi per più di un motivo. In primis sulla situazione reale di tiro. In quei viaggi appresero che le circostanze delle reali situazioni di controllo di tiro, le quali “erano ben più complicate della caccia all'anatra”. Inoltre Wiener e Bigelow si resero conto della reale situazione dell'aviazione americana rispetto alla *Luftwaffe* tedesca: “Il dislivello fra la potenza aerea dell'Asse e gli armamenti americani, migliorati di pochissimo dai tempi della Grande guerra, divenne inoltre manifesto.”. I bombardieri tedeschi sfrecciavano a 500 km all'ora a più di novemila metri di altitudine. Un colpo sparato dall'artiglieria contraerea alleata poteva impiegare più di 20 secondi per raggiungere un aereo nemico. La procedura, inoltre, era goffa e farraginosa, “Una squadra, composta anche da quattordici uomini, aveva l'incarico di individuare l'aereo con dei binocoli, tenerlo costantemente sotto osservazione, trasmettere la sua mutevole posizione ai calcolatori umani, che tratteggiavano grossolani calcoli manuali per stabilire la collocazione dell'aereo in proiezione e li passavano agli artiglieri che, a loro volta, dovevano ruotare le pesanti torrette a manovella per sparare, infine, una raffica di colpi verso il punto d'impatto designato” [Conway, Siegelman, 2005, pp.160-162].

La nozione di macchina ad apprendimento è vecchia quanto la stessa cibernetica. [...] Le caratteristiche lineari del previsore impiegato in un qualsiasi dato momento dipendono da una conoscenza statistica a tempo lungo del l'insieme della serie temporale che desideriamo predire [...] ed è quindi perfettamente possibile ideare un calcolatore che elabori queste statistiche e sviluppi le caratteristiche a tempo breve del previsore sulla base di un'esperienza già acquisita dalla stessa macchina mentre viene impiegata per la previsione ed elaborata automaticamente [Wiener, 1961 pp. 15-16, in Wiener, 1948 trad. it.].

Senza entrare adesso nel dettaglio dell'analisi della matematica delle operazioni lineari e non lineari elaborata da Wiener, si può notare a riguardo del nesso apprendimento-statistica come "il previsore si serve dell'immediato passato del volo dell'aeroplano come di uno strumento per la previsione del futuro mediante un'operazione lineare». Ma questo problema di primo ordine, quello lineare, è «un problema statistico ove la base per l'elaborazione statistica è fornita dal passato a lungo periodo del volo e dal passato di molti altri analoghi voli" [Wiener, 1948, p.224 trad. it]. E la caratteristica fondamentale affinché le macchine apprendano è la retroazione. Scrive Wiener: "In generale, una macchina ad apprendimento funziona mediante feedback non lineare" [Wiener, 1948, p.224 trad. it]. E in questo modo funzionava e ed apprendeva il "*debomber*" [Dyson, 2012 p.133 trad. it.] di Wiener e Bigelow.

Individuata la chiave di volta del comportamento di tutti i sistemi - organici e meccanici - nel servomeccanismo a retroazione negativa, era ora necessario alimentare il predittore con dei dati, rappresentati dalle *performances* passate del pilota. "Il suo [del predittore] algoritmo per la predizione su input statistici dalle prestazioni passate del pilota" (Galison, 1994, p.51 n.28 trad. it.). Lo scopo era creare una macchina via via più perfezionata per calcolare le mosse future del pilota a partire dai suoi schemi di volo caratteristici. La macchina era pensata per correggere via via la sua prestazione in base all'obiettivo. Questa azione era la struttura basilare di ogni comportamento finalizzato.

È possibile introdurre qui un altro elemento, che attraverserà l'intero corso di questo lavoro. "I cibernetici, in ultima analisi, cercarono sempre di costruire cervelli" [Edwards, 1996, p. 240, trad. mia]⁴⁶. Come ha osservato Michael Arbib (1940-), matematico e neuroscienziato, il cibernetico è colui che "cerca di progettare, almeno in linea di principio, macchine di elaborazione dell'informazione che rispecchino il comportamento intellettuale umano o il

⁴⁶ "The cyberneticians always ultimately sought to build brains".

funzionamento cerebrale". L'obiettivo è dunque "il *design* di macchine la cui struttura rispecchia [*parallels*] quella del cervello" [Arbib, 1972, p. 5, trad. mia]⁴⁷, in forme di computazione distribuita e organizzata secondo principi somatotopici [Arbib, 1972, p.5 trad.mia]⁴⁸. Accogliendo la teoria dell'informazione - di cui Wiener fu tra i fondatori - queste strutture assumeranno progressivamente una forma neuromorfica, ispirata a reti neurali e sinaptiche. Come mostra l'archetipo delle macchine cibernetiche, il predittore antiaereo, tali sistemi devono operare su dati per gestire l'incertezza probabilistica che ne caratterizza il funzionamento. Ne è prova, come vedremo, la riflessione sulla torsione probabilistica della fisica che apre *Cibernetica* del 1948, messa in relazione ai sistemi di feedback e input-output delle macchine che la nascente scienza aveva per obiettivo costruire:

La relazione di questi meccanismi con il tempo richiede uno studio accurato. È chiaro, naturalmente, che la relazione ingresso-uscita sia una relazione consecutiva nel tempo e implichi un preciso ordine passato-futuro. Ciò che forse non appare altrettanto evidente è che la teoria degli automi sensibili sia, in realtà, una teoria statistica. Una macchina rientrante nella tecnica delle comunicazioni che funzionasse soltanto per un singolo ingresso ci interesserebbe assai poco. Per essere realmente efficace, essa deve operare in modo soddisfacente per un'intera classe di ingressi, vale a dire deve garantire un funzionamento statisticamente adeguato rispetto alla classe di ingressi che ci si attende di ricevere. La teoria di questa macchina appartiene quindi più alla meccanica statistica gibbsiana che non alla meccanica classica newtoniana [Wiener, 1948, pp.71-72 trad. it.].

È inevitabile, sulla scorta di questa prospettiva, notare come il nucleo centrale dello strumento che Wiener e Bigelow avevano costruito, conteneva *in nuce*, prefigurando e dando forma al tipo di interazione uomo-macchina conseguente, le caratteristiche proprie della tecnologia che sta rivoluzionando la nostra contemporaneità fino a diventare, erroneamente, sinonimo della stessa Intelligenza Artificiale: il *Machine Learning* (ML).

Il Machine Learning può essere ampiamente definito come un insieme di metodi computazionali che utilizzano l'esperienza per migliorare le prestazioni o per fare previsioni accurate. Qui, per

⁴⁷ "We shall refer to this type of study as cybernetics, extending the usage of Norbert Wiener [1948] who defined the subject as "the study of control and communication in man and machine." For our present purpose, we may say that a worker in the field, of cybernetics seeks to design, at least in principle, sophisticated information processing machines, which parallel human intellectual behavior or brain function".

⁴⁸ "“Distributed action-oriented computation in layered somatotopically organized machines”

"esperienza" si intende l'informazione passata a disposizione del *learner*, che tipicamente assume la forma di dati elettronici raccolti e resi disponibili per l'analisi⁴⁹ [Mohri et al., 2022, p.1 trad mia]⁵⁰.

Pur rimanendo sulla superficie dell'algoritmo di apprendimento, il ML si può definire come una tipologia di inferenza statistica su base induttiva, che ha l'obiettivo di trovare correlazioni tra dati, e i cui output, e i pattern individuati, sono accurati in funzione del numero di dati forniti. "Quando l'agente è un computer [o un altro agente artificiale], lo chiamiamo *machine learning*: un computer osserva alcuni dati, costruisce un modello basato sui dati e utilizza il modello sia come ipotesi sul mondo sia come un software che può risolvere problemi"⁵¹ [Russell, Norvig, 2021].

Nel ML - come analizzeremo in seguito ad un livello d'astrazione più vicino alla struttura *hardware* delle componenti del calcolatore e *wetware* del cervello - il feedback correttivo, la retroazione negativa ha una funzione fondamentale, specialmente nelle modalità di *apprendimento supervisionato* e nell'algoritmo di retropropagazione [*backpropagation*] dell'errore. Durante l'addestramento supervisionato il sistema di apprendimento "riceve un esempio, genera un output, e gli viene poi dato un "segnale di supervisione", che dice in quale misura l'output del sistema differisce dall'output corretto. Il sistema usa poi questo segnale per regolare i propri pesi e la propria soglia" (Mitchell, 2021, p. 16 trad. it.). Il modello si propone di anticipare l'obiettivo corretto per un determinato input e successivamente riceve un feedback che lo aiuta a migliorare.

Questo processo avviene sotto la supervisione di un sistema che fornisce all'algoritmo un insieme di dati pre-classificati o etichettati. In questo modo, l'algoritmo è in grado di confrontare le sue previsioni iniziali con i risultati corretti e, attraverso l'analisi di queste

⁴⁹ "Machine learning can be broadly defined as computational methods using experience to improve performance or to make accurate predictions. Here, experience refers to the past information available to the learner, which typically takes the form of electronic data collected and made available for analysis."

⁵⁰ "

⁵¹ "When the agent is a computer, we call it machine learning: a computer observes some data, builds a model based on the data, and uses the model as both a hypothesis about the world and a piece of software that can solve problems». Per una trattazione accurata ma accessibile anche al lettore non specialista sul ML si vedano Mitchell (2019); Larsson (2021, pp. 133-155 trad. it) che ha il merito di analizzare l'apprendimento automatico in relazione ad una più ampia trattazione del carattere *mitologico* dell'Intelligenza Artificiale Generale (AGI) e delle motivazioni epistemologiche, sociopolitiche e tecnologiche che rendono questa prospettiva chimerica; e Domingos (2015), sebbene la sua analisi degli algoritmi sia viziata da una prospettiva quasi messianica intorno ai poteri e alle potenzialità del *Machine Learning*. Una efficace sintesi che non richiede competenze informatiche che spiega anche le diverse modalità di addestramento del ML è quella in Boden (2018, pp. 47-50 trad. it). Per una trattazione maggiormente tecnica si rimanda a Mohri et al. (2018) e al Capitolo V della quarta edizione di quella che è considerata la "bibbia" dello studio dell'Intelligenza Artificiale, *Artificial Intelligence: A Modern Approach* di Stuart Russell e Peter Norvig (2021).

discrepanze, può adattare e modificare i propri parametri interni. Questo continuo aggiustamento e apprendimento gli consente di affinare le sue capacità predittive, migliorando progressivamente le prestazioni nelle previsioni future ⁵².

Il feedback ha un ruolo centrale anche nella *backpropagation*, che analizzeremo anche matematicamente, l'algoritmo di addestramento fondamentale nel sottoinsieme del ML protagonista dello straordinario successo delle ICT, tra cui l'Intelligenza Artificiale contemporanea, il *Deep Learning*. Quest'ultimo, l'apprendimento profondo, è una definizione che si "riferisce all'apprendimento automatico che utilizza più livelli di elementi computazionali semplici e regolabili" [Russell, Norvig, 1921]⁵³. Il termine "*deep*", profondo, indica la profondità e la convoluzione di strati delle reti neurali artificiali⁵⁴, *artificial neural network* (ANN), ispirate alle reti sinaptiche organiche e ai neuroni del cervello umano⁵⁵.

⁵² Il ciclo di correzione a feedback nel *Machine Learning* ad apprendimento supervisionato funziona in questa maniera: "Esso [il processo di apprendimento] richiede in genere un ampio insieme di esempi positivi (per esempio, una serie di numeri 8 scritti da persone diverse) e di esempi negativi (per esempio, una serie di altri numeri scritti a mano, che non includono i numeri 8). Ciascun esempio è etichettato da una persona, che ne indica la categoria: in questo caso 8 oppure non-8. Questa etichetta sarà usata come segnale di supervisione" [Mitchell, 2019, p.17 trad.it.]. In questo contesto, ogni esempio di addestramento è composto da una coppia di input (dati) e output desiderato (etichetta). Il modello viene addestrato cercando di prevedere l'output corretto per ogni input. Quando il modello fa una previsione, questa viene confrontata con l'etichetta reale. Questo ciclo di previsione, valutazione dell'errore e aggiornamento dei pesi costituisce il *feedback* correttivo nel *machine learning* supervisionato.

⁵³ "The term deep learning refers to machine learning using multiple layers of simple, adjustable computing elements".

⁵⁴ L'antecedente nella storia della scienza delle reti neurali artificiali non può non essere identificato nel seminale lavoro di McCulloch e Pitts del 1943 *A Logical Calculus of Ideas Immanent in Nervous Activity*, che sarà ulteriormente analizzato avanti in cui il neurone, e l'attività neuronale del cervello vengono trattati *computazionalmente*, capaci di logiche di input binari. "I valori *vero/falso* della logica vennero fatti corrispondere agli stati di attività *on/off* delle cellule cerebrali e ai singoli stati *0/1* di una macchina di Turing" [Boden, 2018, p. 15 trad it). Su ANN e *Deep Learning* si veda Boden (2018, pp. 79-99 trad. it); Lee e Chen [2021 pp. 24-52 trad. it.] che offrono nel loro *AI 2041*, accanto ad avvincenti scenari di fantascienza verosimile e realistica a breve termine (il 2041 del titolo del loro testo), spiegazioni accessibili alle tecnologie che plasmano e plasmeranno la nostra contemporaneità e il nostro futuro.

⁵⁵ Le reti neurali artificiali, che analizzeremo nel dettaglio più avanti, vengono utilizzate come modello esplicativo da un paradigma di filosofia della mente e di scienza cognitiva: il connessionismo. Quest'ultimo si sviluppa all'interno delle teorie computazionalistiche della mente, ma ritiene «che le computazioni da prendere in considerazione non siano da assimilare a calcoli logici (e non abbiano un carattere "linguistico"), ma debbano essere sviluppate attraverso il modello delle "reti neurali"» [Di Francesco, 2003, p.175]. Queste reti «agiscono in parallelo, e (nelle varianti ortodosse) negano così l'esistenza di un livello intermedio autonomo di rappresentazioni, considerando piuttosto il mentale come emergente a partire dalla serie di "reti" che si auto-organizzano sulla base dell'interazione dell'individuo con l'ambiente» (Di Francesco, 2003, p.19). L'approccio connessionista, a differenza del computazionalismo classico, opera a livello subsimbolico, basandosi su caratteristiche elementari che, prese singolarmente, non hanno un significato globale. "Ogni caratteristica viene rivelata attraverso specifiche microunità cognitive, che contribuiscono a formare un significato emergente dall'insieme, piuttosto che da singoli simboli isolati" (Pessa, 2001, p.98). La cognizione e la rappresentazione «emergono come proprietà globali del tutto, e non esiste al suo interno nulla di simile a rappresentazioni puntuali e costanti manipolate da regole formali, né soprattutto esiste qualcosa di simile a un "linguaggio del pensiero"» (Di Francesco, 2001, p.176). I vantaggi del connessionismo rispetto al calcolo simbolico del computazionalismo classico starebbero nel superare le difficoltà di quest'ultimo di dare ragione ai processi di livello inferiore della cognizione, come la percezione. Sul connessionismo si vedano Stinson, (2019); Buckner, Garson, (2018) e

Prima dell'analisi più approfondita che seguirà – a partire dagli automi di von Neumann e dal *Perceptron* di Rosenblatt [1958] – è utile introdurre brevemente il concetto di rete neurale. Le reti neurali sono sistemi composti da unità (neuroni o nodi) che operano come dispositivi ingresso-uscita: l'attivazione di ciascuna unità dipende dal suo potenziale di attivazione, determinato dagli input ricevuti. Le unità sono collegate tra loro tramite connessioni dotate di peso e direzione, che definiscono la forza con cui l'attivazione si trasmette da un nodo all'altro [cfr. Pessa, 2001, p. 99]. Nel *deep learning* i diversi strati computazionali si dispongono tra un livello di input e un livello di output [Lee, Chen, 2021, p. 41, trad. it.], e i valori dei pesi determinano la trasformazione dei segnali da uno strato all'altro.

Tra le procedure di apprendimento, un ruolo centrale è svolto dalla retropropagazione [*backpropagation*], una delle tecniche più efficienti per reti multistrato di unità neuronali, apprezzata anche per la sua semplicità [LeCun, 1988]. Essa consente alla rete di adeguare i propri pesi e *bias* in base all'errore tra l'output previsto e quello effettivo: l'errore viene propagato all'indietro attraverso i livelli, aggiornando i parametri e migliorando progressivamente le prestazioni della rete.

La stessa struttura della macchina di Wiener – base della futura cibernetica – può essere descritta come un modello statistico, addestrato su dati reali e capace di correggersi tramite retroazione negativa per produrre output predittivi. Una logica che, come abbiamo appena visto, si ritrova anche negli algoritmi di *machine learning* e *deep learning* che alimentano le odierne tecnologie digitali pervasive.

Dal punto di vista analizzato, organismo e meccanismo, uomo e macchina, ricadono entrambe sotto quella definizione di apprendimento che può essere considerata *standard* anche nel caso di entità artificiali:

Si dice che un programma informatico apprenda dall'esperienza *E* rispetto a una certa classe di compiti *T* e a una misura di prestazione *P*, se le sue prestazioni nei compiti in *T*, misurate da *P*, migliorano con l'esperienza *E* [Mitchell, 1997, p.2 trad. mia]⁵⁶.

Smolensky (1989) Quest'ultimo anche per la trattazione e la proposta del *Subsymbolic Paradigm*. Per un'analisi del connessionismo legata alla logica della computabilità, si veda Frixione 1992. (2001). Sul rapporto-differenza tra il paradigma subsimbolico e l'intelligenza artificiale cognitiva classica si rimanda a Clark, 1989 e Parisi, 1989.

⁵⁶ "A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P*, if its performance at tasks in *T*, as measured by *P*, improves with experience *E*".

“L'apprendimento è una forma molto complessa di retroazione, e incide non soltanto sull'azione individuale ma anche sul modello dell'azione. Essa è anche un mezzo per rendere il comportamento meno subordinato alle esigenze dell'ambiente” [Wiener, 1950, p.84 trad. it]. L'essere umano, dal punto di vista del *servomeccanismo*, del *feedback* e del comportamento, non differisce da macchine che Wiener aveva preso in considerazione nei suoi lavori sulla cibernetica, come ad esempio, come vedremo, quella per giocare a dama di A. L. Samuel dell'IBM [1959], “un programma che permette a un calcolatore di giocare a dama, e questo calcolatore impara, o almeno sembra che impari, a migliorare il suo gioco con l'esperienza” [Wiener, 1964, p.20 trad. it.].

L'immagine della natura essenziale degli enti organici e artificiali come *black box* a retroazione negativa, non è un'immagine socio-culturalmente neutra. È direttamente figlia dei suoi tempi e del progetto di costruzione di armi capaci di ribaltare la superiorità tecnica del nemico. “È un'immagine delle relazioni umane totalmente ancorata nel progetto e nella fabbricazione di servomeccanismi in tempo di guerra ed estesa, nella generalizzazione definitiva, a un universo di monadiche scatole nere” [Galison, 1994, p.80 trad. it.].

Le condizioni in cui è stato piantato il seme cibernetico, la costruzione del predittore, hanno lasciato probabilmente traccia sul modo con cui si configurano oggi il digitale e le ICT che rappresentano anch'ess ambiti in cui “Il significato culturale di concetti e pratiche, ritengo, è indissolubilmente legato alla loro genealogia” [Galison, 1994, p.79 trad. it.]⁵⁷.

La forma della nostra contemporaneità è plasmata, rivoluzionata, da tecnologie che hanno il loro antecedente nella scienza originata dagli studi sul predittore antiaereo. E in questa stessa origine possiamo coglierne la traccia culturale che ha lasciato sin oggi. “Per comprendere il significato culturale specifico dei dispositivi cibernetici è necessario rintracciarli

⁵⁷ Peter Galison ha esplorato il concetto di significato culturale, che ha animato la sua analisi della cibernetica di Wiener in *The Ontology of the Enemy* del 1994, anche un anno prima in *The Cultural Meaning of Aufbau* [Galison, 1993], riguardo l'uso e il senso assunti dal termine tedesco eponimo. Galison analizza i tre principali picchi d'uso del termine "Aufbau" nel contesto culturale e politico tedesco: dopo la Prima Guerra Mondiale (1919-1927), come simbolo della ricostruzione razionale e modernista; durante l'ascesa del Nazismo (1934-1937), conteso tra ideologie contrapposte di destra e sinistra; e nel dopoguerra (1945-1947), legato alla rinascita democratica e internazionale. "Aufbau" riflette un'idea di trasformazione radicale che attraversa architettura, educazione e politica, incarnando il rinnovamento culturale totale. Nel neopositivismo, Carnap lo utilizza in *Der logische Aufbau der Welt* (1928) per indicare un progetto epistemologico di ricostruzione su base razionale del mondo, L'uso del termine non avrebbe avuto lo stesso significato senza la genealogia che lo ha plasmato culturalmente. Il concetto di “significato culturale” di Galison è qui particolarmente utile come lente analitica per indagare l'impronta lasciata dall'innesto della cibernetica sulle traiettorie di sviluppo delle tecnologie digitali, sul loro rapporto con l'essere umano, e sulla visione dell'uomo che ne è scaturita e che continua ancora oggi a plasmarle.

precedentemente nella visione dei tempi di guerra del pilota-come-servomeccanismo” [Galison, 1994 p,79 trad. it.]

La prima mossa di Wiener e dei suoi collaboratori fu quella di incorporare nella macchina il pilota, inteso come elemento randomizzato, a causa delle limitazioni materiali e fisiologiche che influenzavano la predizione. «Dal corpo fummo noi, più in generale, a essere considerati così – noi umani – le cui intenzioni potevano essere viste come nient’altro che entità autocorreggenti nelle scatole nere» [Galison, 1994, p. 79, trad. it.]. La fusione, forse stereotipata, di organico e meccanico in un’unica entità – attribuita alla cibernetica e, a livello di cultura popolare, incarnata nel suo esito più celebre, il *cyborg*, “un organismo cibernetico, un ibrido di macchina e organismo” [Haraway, 1985] – deriva probabilmente da una lettura semplificata delle teorie wieneriane [Leonzi, Ciofalo, 2017, p. 20]. Tuttavia, i nodi critici del rapporto tra dimensione umana e macchine, tra biologico e artificiale, tra sistema nervoso e computer, restano oggi di estrema attualità.

L’integrazione che Wiener realizzò nel predittore – e ancor prima, nello studio della fisiologia propriocettiva dello “pseudo-pilota” utilizzata per generare i dati con cui addestrare la macchina – avveniva a un livello più basilare della semplice, e quasi fantascientifica, fusione tra naturale e artificiale. La ragione dello sfumare dei confini tra queste due dimensioni risiedeva infatti in un principio più fondamentale: “era l’impulso comportamentista a focalizzarsi su ampie classi di azioni, e a fare così sulla base dell’input e dell’output” [Galison, 1994, p. 61, trad. it.], che spinse Wiener ad assumere il servomeccanismo come criterio centrale, indipendentemente dalla natura organica o meccanica del sistema in esame. In questo senso, il corpo – come mostrano anche gli esempi tratti dalla controcultura – la sua stessa negazione, o la torsione in senso biologista delle strutture computazionali, tornano a occupare una posizione cruciale.

L’interazione uomo-macchina, dalle macchine calcolatrici analogiche come l’*Analizzatore Differenziale* di Vannevar Bush⁵⁸, usata da Wiener per calcolare traiettorie dei

⁵⁸ L’Analizzatore Differenziale (*Differential Analyzer*), perfezionato da Vannevar Bush nel 1931 al MIT, era una macchina analogica imponente, pesante circa 100 tonnellate, costituita da aste rotanti, dischi e ingranaggi meccanici. Questa macchina era progettata per risolvere equazioni differenziali, che sono equazioni matematiche utilizzate per descrivere le relazioni tra diverse grandezze fisiche e i loro tassi di variazione nel tempo e nello spazio, come velocità e accelerazione. Una versione precedente del 1927 aveva conquistato la prima pagina del *New York Times* che recitava: «La “macchina pensante” fa matematica superiore: risolve equazioni che richiedono mesi a un essere umano» (Owens, 1986, trad. mia). L’Analizzatore rappresentava queste grandezze non come numeri digitali, ma come analoghi fisici, permettendo di generare curve che descrivevano il comportamento dinamico dei sistemi fisici in modo molto più rapido rispetto ai calcoli manuali, riducendo notevolmente il tempo necessario per risolvere problemi matematici complessi. (Bush, 1931; Cfr. Gleick, 2011, pp.161-163, trad. it.).

velivoli, sino ai *Large Language Models*⁵⁹ (LLM) odierni si è sviluppata lungo la dorsale di *imitazione – riduzione-integrazione/blurring* umano/macchina – *riproduzione/reificazione* in dati e previsione mediante algoritmi. “Dopo aver cercato, quindi di non solo esorcizzare il corpo chiudendolo tra parentesi, nell’ipotesi di riuscire a costruire sistemi simbolici del tutto privi di riferimento fisico, attraverso la mediazione della cibernetica” adesso, la forma più attuale assunta dalle ICT, l’Intelligenza Artificiale (AI) si “è orientata verso l’interazione comunicativa tra esseri umani e macchine. Tale interazione in un primo tempo si è manifestata come integrazione, in cui la macchina era sotto il controllo dell’operatore umano e poi, con l’avvento dei dati prodotti dalle azioni umane registrate, ha assunto sotto di sé la capacità di prendere decisioni” [Numerico, 2021, p.26]. L’analisi delle condizioni culturali in cui si è sviluppata questa forma di agentività decisionale è imprescindibile per una sua comprensione critica.

Ma il far interagire la macchina/algoritmo con i dati di un sistema che ne generalizza i comportamenti al fine di prevederne le azioni future (nucleo del ML e del DL); questo prevedere statisticamente gli output a partire da serie storiche di input, come abbiamo visto, era già al centro del progetto di Wiener. Il quale si è sviluppato come tale portando con sé una serie di significati culturali propri del tempo in cui è nato, *in primis* la visione “dell’Altro. Noi siamo veramente, in questa visione del mondo, come scatole nere con input ed output e nessun accesso alla nostra o all’altrui vita interiore” [Galison, 1994, p.71 trad. it]. La *black box* rappresentata dall’altro diventa il modello per l’analisi del Sé. Lo stesso servomeccanismo a retroazione- come esplicitamente ammesso da Wiener nella scelta di battezzare *Cibernetica* la scienza che sullo studio di questi meccanismi si fondava in omaggio ⁶⁰ - è diventato il concetto essenziale in un contesto in cui la lotta contro il nemico “ridusse il comportamento umano a mosse di inseguimento, fuga e inganno; e introdusse una nuova classe di armi autoregolanti. È in questo contesto specifico che fu forgiata l’identità di intenzione e autocorrezione” [Galison, 1994, p.78 trad. it].

⁵⁹ Un *Large Language Model* (LLM) è un’intelligenza artificiale, o meglio un algoritmo statistico computazionale auto-supervisionato o semi-supervisionato, addestrato su ampi dataset di testi per comprendere e generare linguaggio simile a quello umano. L’intelligenza artificiale generativa si riferisce a sistemi di intelligenza artificiale progettati per creare nuovi contenuti, come testo, immagini o musica, imparando schemi dai dati esistenti.

⁶⁰ Scrive Wiener: “La scelta di questo termine [*Cibernetica* e il suo derivarlo dalla parola greca *κυβερνήτης*, timoniere] è dovuta al riconoscimento che abbiamo inteso dare al fatto che il primo significativo scritto sui meccanismi a feedback è un articolo, sui regolatori (*governors*) pubblicato da Clerk Maxwell nel 1868” [Wiener, 1948, p.35 trad. it].

Il livello d'astrazione⁶¹ al quale considerare le caratteristiche fondamentali di ogni agente, il feedback e la natura a scatola nera, trova un filo rosso, nell'età della rivoluzione digitale, con quanto sa ogni studente alle prime armi di un corso di *computer science*:

Nel primo giorno tipico di un corso introduttivo di informatica, viene spiegato il concetto di astrazione. Gli studenti imparano che i sistemi possono essere descritti come scatole nere, definite precisamente dai loro input, output e dalla relazione tra di essi. Le proprietà desiderabili di un sistema possono quindi essere descritte in termini di input e output: l'interno del sistema e la provenienza degli input e degli output sono stati astratti [Selbst et al., 2018 trad. mia]⁶².

Inoltre, a trattare questi sistemi a scatola nera, vi sono programmi di *Machine Learning* e di *Deep Learning* che a loro volta sono *black box* a loro volta. Quello che avviene tra le profonde convoluzioni degli strati di reti neurali degli algoritmi di apprendimento è inconoscibile. "Il deep learning è impenetrabile persino per i suoi programmatori perché tipicamente utilizza migliaia o milioni di connessioni che interagiscono tra loro in modi complessi" [Liao, 2020, p.7]⁶³. Queste scatole nere computazionali a scopo predittivo possono funzionare perché i sistemi da cui traggono i dati per prevederne il comportamento futuro sono stati "ridotti" anch'essi a *black box*. Gli algoritmi funzionano nelle situazioni in cui vige quello che Gigerenzer, direttore emerito del *Max-Planck-Gesellschaft zur Förderung der Wissenschaften*, ha definito il *principio del mondo stabile*:

Gli algoritmi complessi funzionano al meglio in situazioni ben definite, stabili, in cui sono disponibili grandi quantità di dati. L'intelligenza umana, invece, si è evoluta in modo da gestire l'incertezza a prescindere dalla quantità di dati disponibili [Gigerenzer, 2022].

Lo stesso avviene nella produzione di output predittivi da parte dei sistemi computazionali: essi funzionano meglio quando la realtà su cui operano viene ridotta a una struttura generalizzabile, assimilabile a un servomeccanismo a feedback negativo. Questo

⁶¹ Un *Livello di Astrazione* (LoA) è un insieme non vuoto di variabili tipizzate interpretate (osservabili) che costituiscono l'accesso, in relazione al nostro scopo epistemico, ad un oggetto (il sistema) per la costruzione di un modello del sistema in questione. Più LoA costituiscono un'*interfaccia*. [Floridi, 2008]

⁶² "On the typical first day of an introductory computer science course, the notion of abstraction is explained. Students learn that systems can be described as black boxes, defined precisely by their inputs, outputs, and the relationship between them. Desirable properties of a system can then be described in terms of inputs and outputs alone: the internals of the system and the provenance of the inputs and outputs have been abstracted away".

⁶³ "Deep Learning is impenetrable even to its programmers because it typically employs thousands or millions of connections that interact with one another in complex ways".

livello di astrazione, pur riduttivo, si affermò come realistico in un contesto in cui - anche per ragioni socio-economiche - non poteva essere altrimenti. “Faccia a faccia con un’altra persona, senza alcun modo per evitare la profondità e l’ambivalenza dell’interazione umana, la retroazione poteva sembrare [...] una rappresentazione ridicolmente semplicistica dell’intenzionalità” [Galison, 1994, p. 78, trad. it.]. Oggi, come negli anni del secondo conflitto mondiale, gli algoritmi eludono sempre più spesso il contatto diretto “faccia a faccia”, poiché l’azione e l’interazione umana sono ormai pervasivamente mediate dalle ICT.

Questa costruzione di scatole nere, in cui “sistemi opachi di *black box* [...] prendono decisioni importanti sulla vita delle persone” [O’Neil, Gunn, 2020, p. 239]⁶⁴ apre questioni cruciali di interpretabilità, spiegabilità e fiducia [Liao, 2020, p. 7]⁶⁵, che sono sia etiche e sia epistemologiche⁶⁶. Oggi alla predizione algoritmica vengono affidate decisioni che riguardano l’accesso al welfare, al credito, alla selezione per un impiego, alla valutazione del rischio di recidiva criminale, e molto altro [O’Neil, 2016]. Ma gli algoritmi incidono anche nella sfera della quotidianità e persino in quella affettiva: dalla scelta di un ristorante a quale film guardare, dalla musica da ascoltare fino alla selezione di un potenziale partner.

Per affrontare - o almeno attenuare - le criticità in termini di trasparenza, *accountability* e controllo democratico della *Black Box Society* [Pasquale, 2015] - la società in cui decisioni cruciali di natura sociale ed economica vengono affidate ad algoritmi e sistemi automatizzati il cui funzionamento interno resta opaco o incomprensibile ai più - è fondamentale risalire alle radici culturali che ne hanno plasmato le forme e le logiche.

Come nel caso dell’algoritmo statistico del predittore antiaereo, la *learning machine* progettata da Wiener, “i sistemi di apprendimento automatico sono progettati per giungere a una generalizzazione a partire da un ampio insieme di casi usati come addestramento, per poi classificare in modo corretto nuovi dati non inclusi nel *dataset* di *training*” [Crawford, 2021, trad. mia]⁶⁷. Questi sistemi eseguono una induzione, imparando da esempi specifici con lo scopo di decidere cosa cercare in ulteriori esempi che gli vengono forniti. Ma il loro processo di

⁶⁴ “Opaque black box systems that make important decisions about people’s lives”.

⁶⁵ “Deep learning is a black box that raises issues such as interpretability, explainability, and trust”

⁶⁶ La trasparenza algoritmica è una questione centrale all’interno del dibattito intorno alle problematichità dell’Intelligenza Artificiale. Luciano Floridi propone un “quadro generale costituito da cinque principi per l’IA etica” [Floridi, 2022, p.93 trad. it.]. Il quadro dei quattro principi fondamentali, comunemente usati in ambito bioetico, «*beneficenza, non maleficenza, autonomia e giustizia*» va ampliato con un nuovo principio *abilitante* per l’IA, il principio di *esplicabilità*, che include “sia il senso *epistemologico* di *intelligibilità* (come risposta alla domanda: “Come funziona?”) sia quello *etico* di *responsabilità* (*accountability*) (come risposta alla domanda “Chi è il responsabile del modo in cui funziona?”)” [Floridi, pp.95-96, p.101 trad. it.].

⁶⁷ “Machine learning systems are designed to be able to generalize from a large training set of examples and to correctly classify new observations not included in the training datasets”.

generalizzazione e *clusterizzazione* statistico-induttiva, spesso, come nel caso delle caratteristiche somatiche (misurazione del cranio, colore della pelle e così via), si attua perché tecnicamente più fattibile per gli algoritmi [Crawford, 2021, pp.123-133].

I *bias* discriminatori sono dovuti alla forma tecnica dell'algoritmo⁶⁸. Si verifica una "produzione centralizzata dell'identità, guidata dalle tecniche di apprendimento a disposizione" [Crawford, 2021, p.133 trad. mia]⁶⁹.

Allo stesso modo, il contesto bellico - con le sue necessità e con lo stato dell'arte scientifico-tecnologico - aveva universalizzato e innalzato la visione del mondo, umano e naturale, di Wiener: "un'estensione generalizzata, persino metafisica, della lotta epocale tra il nemico implacabile dal cielo e il predittore antiaereo degli Alleati che effettua calcoli e che porta avanti la battaglia dal suolo" [Galison, 1994, p. 80, trad. it.], divenuta spiegazione delle caratteristiche di ogni entità organica e meccanica.

Per essere efficace, la predizione deve basarsi su dati misurabili, indipendentemente dal sistema di riferimento. "L'appartenenza a un cluster di persone che sono tenute assieme dalla somiglianza dei loro comportamenti o da altre variabili prescritte deve escludere altre caratteristiche che il modello non prende in esame" [Numerico, 2021, p. 156]. Le caratteristiche individuali - o, per usare il livello di astrazione impiegato da Wiener, quelle che rimangono fuori dall'analisi funzionale - vengono escluse. In questo processo, i giudizi di astrazione e le riduzioni a cluster o schemi misurabili diventano *biased*, "perché non possono entrare nella singolarità di un caso, pena l'inefficienza" [Numerico, 2021, p. 156].

La varietà delle caratteristiche e dei comportamenti umani rischia di essere ridotta a fattori tecnici, i quali sono a loro volta profondamente determinati dalle condizioni socio-

⁶⁸ È interessante la genealogia che Crawford propone del concetto di *bias*, in quanto ne mette in luce sia la natura statistica sia il significato culturale che via via ha assunto. "La genealogia storica del termine "bias" è molto più recente. Compare per la prima volta nella geometria del XIV secolo, dove si riferisce a una linea obliqua o diagonale. Nel XVI secolo aveva già acquisito un significato simile a quello oggi corrente, ossia "pregiudizio indebito". Negli anni 1900 "bias" sviluppò un'accezione più tecnica in statistica, dove indica le differenze sistematiche tra un campione e la popolazione, quando il campione non riflette in modo autentico l'insieme. È da questa tradizione statistica che il campo del *machine learning* trae la propria comprensione di "bias", in relazione a un insieme di altri concetti: generalizzazione, classificazione e varianza" ["The historical lineage of "bias" as a term is much more recent. It first appears in fourteenth-century geometry, where it refers to an oblique or diagonal line. By the sixteenth century, it had acquired something like its current popular meaning, of "undue prejudice." By the 1900s, "bias" had developed a more technical meaning in statistics, where it refers to systematic differences between a sample and population, when the sample is not truly reflective of the whole.³⁴ It is from this statistical tradition that the machine learning field draws its understanding of bias, where it relates to a set of other concepts: generalization, classification, and variance", Crawford, 2021, p.135].

⁶⁹ "Diversity grow out of this kind of centralized production of identity, led by the machine learning techniques that were available to the team".

culturali in cui vengono prodotti. Ne vedremo esempi nell'analisi della storia dell'implementazione dei primi calcolatori.

L'aver individuato nel feedback negativo il fulcro dell'analisi di ogni tipo di sistema ha prodotto un'altra caratteristica che è possibile riscontrare nelle tecnologie algoritmiche contemporanee. Wiener, Bigelow e Rosenblueth nel loro saggio del 1943 avevano dato una definizione dell'azione guidata da uno scopo e, come già visto, nella loro analisi "il comportamento teleologico diventa sinonimo di comportamento controllato da retroazione negativa" [Rosenblueth et al., 1943, p.104 trad. it.]. Ristretto il campo a questa definizione di comportamento, e messa in relazione al feedback, "la teleologia non è opposta al determinismo, ma alla non-teleologia" [Rosenblueth et al., 1943, p.104 trad. it.]. Un sistema, teleologico o non-teleologico sono deterministici soltanto quando il dominio cui appartiene il comportamento del sistema in questione è suscettibile di considerazione deterministica. "Il concetto di teleologia ha in comune solo una cosa con il concetto di causalità: un asse dei tempi" [Rosenblueth et al., 1943, p.105 trad. it.].

La retroazione come cuore del comportamento, cioè, porta a una riformulazione del concetto di causalità. Alla relazione causa-effetto [o stimolo-risposta] si sostituisce una "causalità circolare" [Heims, 1991, p.19 trad. it.], come nei casi delle analisi sulla propriocezione di Wiener. Una azione come prendere un bicchiere d'acqua, regolata dal feedback negativo di sensazioni cinestetiche verso lo scopo ha natura causale circolare perché "la posizione del braccio e della mano in un determinato momento è data delle informazioni in entrata per l'azione del momento successivo. Se il circuito è integro, il processo risulta regolare" [Heims, 1991, p.19 trad. it.]

Nelle scienze naturali, ma si potrebbe risalire sino al pensiero greco, "una causa A produce un effetto B" (Ivi), "le forze causano accelerazione". Queste formulazioni hanno condotto ad analisi matematiche semplici (lineari)" [Heims, 1991, p.19 trad. it.].

Ma, come abbiamo visto, lo scopo di Wiener era affrontare, con il suo metodo matematico-statistico, problemi non passibili di analisi lineare [Wiener, 1942]. Una volta individuata nella retroazione negativa la chiave per la previsione (non lineare) delle traiettorie generate dal comportamento del pilota inserito nel sistema umano-meccanico, anche il concetto classico di causalità subisce una riformulazione. "Con la causalità circolare, A e B sono reciprocamente causa ed effetto l'uno dell'altro. Inoltre, non solo A influenza B, ma attraverso B agisce anche su sé stessa [...] Significava che A non può agire su B senza essere essa stessa influenzata" [Heims, 1991, pp. 26-27, trad. it.].

Prima ancora della pubblicazione dell'articolo scritto con Wiener e Bigelow, Rosenblueth aveva presentato le sue riflessioni sui concetti di comportamento, teleologia e causalità in occasione di un incontro dal titolo *Seminario sull'inibizione mentale*, tenutosi dal 13 al 15 maggio 1942 al Beekman Hotel di New York. L'evento, patrocinato e finanziato dalla Macy Foundation - la stessa che avrebbe poi sostenuto materialmente e organizzativamente la serie delle dieci conferenze interdisciplinari note come "*Macy Conferences*" (1946-1953), tappa fondamentale, tra l'altro, per la nascita delle scienze cognitive ⁷⁰ - vide la partecipazione di figure di primo piano come John von Neumann (1903-1957), Warren McCulloch (1898-1969), Claude Shannon (1916-2001), J.C.R. Licklider (1915-1990), Gregory Bateson (1904-1980), Lawrence Kubie (1896-1973), Walter Pitts (1923-1969) e Margaret Mead (1901-1978). Quest'ultima ne fu talmente entusiasta da ricordare: "Non mi accorsi di essermi rotta un dente fino a quando il seminario non fu finito" [Heims, 1991, p.18 trad. it.]. Questo incontro va sottolineato per la sua importanza storica: rappresentò infatti uno dei momenti originari dell'evoluzione della cibernetica come paradigma scientifico - e non solo - a vocazione interdisciplinare.

Di quanto detto a proposito di feedback, causalità e non-linearità, si può rinvenire traccia nell'impatto odierno degli algoritmi predittivi. L'introduzione di queste tecnologie, con la nozione di causalità non lineare che ne costituisce l'origine, ha trasformato l'esperienza umana del tempo. Helga Nowotny - già presidente del *Consiglio Europeo della Ricerca (ERC)* - osserva come "oggi [nel tempo in cui gli algoritmi hanno fatto irruzione nel tempo sociale] la linearità della nostra esperienza temporale si è rotta" [Nowotny, 2022, p. 46, trad. it.]. La causalità della retroazione negativa - l'azione di A su B che, a sua volta, retroagisce su A - comporta infatti una trasformazione della scansione temporale passato-presente-futuro. "Noi non lasciamo affatto il presente, ma lo espandiamo. La macchina del tempo digitale porta il passato in un presente in cui influenza l'esperienza del momento" [Nowotny, 2022, p. 46, trad. it.].

Gli algoritmi, per funzionare a livello inferenziale e tecnologico, hanno bisogno di due condizioni strettamente interrelate, che sopra abbiamo definito come *principio del mondo stabile*: "quello che è accaduto in passato si ripeterà e quello che fanno persone categorizzate come simili ad altre tenderà a estendersi a tutti gli altri membri dello stesso cluster" [Numerico, 2021, p. 170]. La riduzione di ogni sistema a schemi di retroazione, il concetto di causalità che vi è sotteso e la possibilità che, in questo modo, i "meccanismi teleologici" - organismi, macchine

⁷⁰ Per la storia delle *Macy Conferences*, che getta luce sui vari protagonisti del movimento cibernetico, e dei loro itinerari scientifici e personali, da una prospettiva che tiene conto delle istanze socio-politiche in gioco, si rimanda all'essenziale Heims, 1991. Per la trascrizione dei singoli incontri e degli interventi si veda Pias, 2017.

e "sistemi combinati macchina-organismo" - possano essere "suscettibili di studio sistematico", così da "estendere considerevolmente il campo dell'applicazione della scienza esatta" [Heims, 1991, pp. 18-19, trad. it.], trovano un analogo nella tendenza riscontrabile oggi nel comportamento dei più potenti tra quelle "macchine culturali" [Finn, 2017, p. 25, trad. it.] che sono gli algoritmi predittivi.

Gli algoritmi predittivi inizialmente lavorano come *storici*: studiano la storia per individuarne gli schemi. Poi diventano *profeti*: concepiscono formule matematiche che spiegano questi schemi, poi testano queste formule su dati storici conservati a quello scopo e infine le usano per fare previsioni sul futuro. È per questo che Amazon, Google, Facebook e tutti gli altri stanno raccogliendo i vostri dati per nutrire i loro algoritmi: *vogliono trasformare il vostro passato nel vostro futuro*⁷¹ [Lepore 2018, trad. mia].

Disinnescare i *bias*, le storture, le iniquità che le tecnologie computazionali che le tecnologie algoritmiche amplificano con la potenza computazionale oggi disponibile, segnata dalla legge di Moore⁷², significa andare alla loro origine per comprenderne le forme e la struttura tecnica e culturale con cui plasmano l'"eterno presente" al quale sembrano condannarci.

La visione cibernetica di una comprensione unificata – biologica e computazionale – del mondo non ci ha mai lasciati e continua a riapparire nelle metafore tecniche e critiche che usiamo per manipolare e comprendere i sistemi computazionali" [Finn, 2016, p. 23, trad. it.]. Riconoscere la matrice culturale - i semi piantati in un determinato contesto che ha dato origine a uno specifico sviluppo tecno-scientifico - significa anche riconoscere come le tecnologie computazionali e algoritmiche si sviluppino "producendo cultura a livello macro-sociale nello stesso momento in cui producono oggetti, processi ed esperienze culturali" [Finn, 2016, p. 25,

⁷¹ "Predictive algorithms start out as historians: they study historical data to detect patterns. Then they become prophets: they devise mathematical formulas that explain the pattern, test the formulas against historical data withheld for the purpose, and use the formulas to make predictions about the future. That's why Amazon, Google, Facebook, and everyone else are collecting your data to feed to their algorithms: they want to turn your past into your future".

⁷² La *Legge di Moore* è quella formulata da Gordon Moore, cofondatore di Intel, per cui "La complessità di un microcircuito, misurata ad esempio tramite il numero di *transistor* per *chip*, raddoppia ogni 18 mesi (e quadruplica quindi ogni 3 anni)". [Moore 1965]. Lungi da essere una "legge di natura", rappresenta una generalizzazione che segnala l'aumento esponenziale di potenza computazionale e il corrispettivo diminuire dei costi della tecnologia che la implementa. La Legge di Moore è divenuta la base metafisica per la "fede" di fautori dell'avvento della Singolarità tecnologica come Ray Kurzweil, imprenditore, inventore e dirigente di spicco di Google, [Kurzweil, 1990, 1999, 2005].

trad. it.]. Questo riconoscimento, accanto all'analisi epistemologica, è alla base di ogni prospettiva critica che non voglia condannarsi alla sterilità.

3. Rumore, entropia, informazione: la cosa migliore dai tempi degli psichedelici

a. Tabula logica elettrificata: pericoli matematici, statistica e relè

Dopo questo necessario excursus su alcuni aspetti della computazione contemporanea della riflessione intorno alle ICT e al loro impatto, - che sarà approfondito lungo tutto il corso di questo lavoro- è necessario riprendere le fila del discorso intorno all'origine di elementi chiave per comprendere l'era dell'informazione, e rispondere così alle domande "controculturali" posta in apertura di questa analisi.

Il Progetto D. I. C 5980, il piano di Wiener di costruire una macchina utile allo sforzo bellico, che permettesse agli Alleati di ridurre il *gap* tecnico con la macchina bellica nazista, ebbe come nucleo non soltanto l'individuazione nel servomeccanismo a *feedback* negativo del carattere chiave per lo studio scientifico e la trattazione matematica di ogni sistema. Sin nella sua struttura teorico-matematica dall'inizio lo studio intorno al metodo statistico di previsione, come si può desumere dal rapporto *Statistical Method of Prediction in Fire Control*[Wiener, 1942]⁷³, si incardinava intorno ad un altro concetto, una delle chiavi per comprendere la nostra epoca rivoluzionata dalle ICT, quello di *informazione*.

Le idee di Wiener, in cui culminavano i suoi studi compiuti prima della guerra intorno al moto browniano e alla teoria ergodica⁷⁴, si concretizzavano nella "leggendaria " (per la sua difficoltà) [Gleick, 2011, p.218 trad. it.] monografia dal titolo *Extrapolation, Interpolation and Smoothing of Stationary Time Series* [1942b], conosciuta anche, a causa del colore della carta

⁷³ "Furthermore, aircraft which attempt to fly straight courses do not precisely do so, for a variety of reasons. It is also clear that, whatever the path of the enemy aircraft, the inputs to the predictor consist of the true signals, due to the actual motion, upon which are superposed "noise" due to tracking errors"

⁷⁴ Norbert Wiener fu profondamente influenzato dal Teorema Ergodico di G.D. Birkhoff (1931), che considerava cruciale per affrontare le sfide della meccanica statistica *gibbsiana*, validando l'ipotesi ergodica: le medie temporali equivalgono alle medie d'insieme in sistemi dinamici osservati a lungo. "Ergodico" indica sistemi in cui le proprietà statistiche possono essere dedotte dall'evoluzione temporale. Per l'importanza, nell'evoluzione del pensiero di Wiener, anche a livello della visione del mondo "filosofica" wieneriana dello studio di questi aspetti della matematica probabilistica, si veda Masani, 1990, pp.139-160. Per le opere di Wiener intorno a questi temi, commentate dal punto di vista matematico, si rimanda a Masani (ed.), 1979, Vol. I pp. 435-758.

con cui era litografato e della difficoltà della matematica contenuta⁷⁵, come *Yellow Peril*, Pericolo Giallo.

In quest'opera, inizialmente coperta dal segreto militare ed autorizzata alla pubblicazione solo nel 1949, Wiener si pone l'obiettivo di unire due campi che, sebbene "condividano una completa unità metodologica naturale", si sono tuttavia originati e sviluppati "da tradizioni totalmente distinte e si differenziano profondamente per vocabolario e formazione del personale". Questi due campi sono "l'analisi delle serie temporali in statistica e l'ingegneria delle comunicazioni" [Wiener, 1942b]⁷⁶. La mossa di Wiener è quella di sviluppare un metodo, in concordanza con i risultati che si otterranno mediante le analisi propriocettive dei sistemi di addestramento del predittore, di natura probabilistica, "per prevedere il futuro a partire da dati rumorosi, incerti e corrotti relativi al passato" [Gleick, 2011, p.218 trad. it.]

L'analisi prende le mosse dalla determinazione dei coefficienti di autocorrelazione e di spettro correlato. Data una sequenza di numeri $\{a(n)\}$, dove $a(n)$ è il valore al tempo $t = n$, i coefficienti di autocorrelazione $\phi(m)$ per l'intervallo m sono definiti come il valore medio, considerato su tutto il valore temporale, del prodotto di uno di questi numeri con i suoi m -esimi vicini. Matematicamente, Wiener li definisce come coefficienti di auto-correlazione (*auto-correlation coefficients*) [1942b, p.4], resi con l'equazione:

$$\phi(m) = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N a(n)a(n+m)$$

[1942b, p. 4, eq. 0.4215]

Questa relazione è quella che cattura la dipendenza statistica tra i valori della sequenza, consentendo così di evidenziare gli schemi ricorrenti e le regolarità nella serie. È uno strumento cruciale per distinguere le componenti coerenti da quelle casuali.

Lo spettro associato, il "*periodogram*" [1942b, p.6] invece, è definito tramite la *trasformata di Fourier*⁷⁷ dei coefficienti di autocorrelazione. Questa trasformazione

⁷⁵ Per l'analisi dell'astrusa matematica (*abstruse mathematics*) del Pericolo Giallo wieneriano si veda Masani pp. 182-195, dove si prende, inoltre, in esame la relazione tra le idee di Wiener e quelle di Kolmogorov intorno alle idee intorno alla previsione e alla probabilità esposte del matematico. La relazione è narrata dallo stesso Wiener nella sua autobiografia [Wiener, 1953b, pp.261-263].

⁷⁶ "This book represents an attempt to unite the theory and practice of two fields of work which are of vital importance in the present emergency, and which have a complete natural methodological unity, but which have up to the present drawn their inspiration from two entirely distinct traditions, and which are widely different in their vocabulary and the training of their personnel. These 'two fields are those of time series in statistics and of communication engineering." Wiener, 1942b, p.1.]

⁷⁷ La Trasformata di Fourier, introdotta da Joseph Fourier in *Théorie analytique de la chaleur* (1822), è uno strumento matematico che scompone un segnale complesso in una somma di oscillazioni elementari, dette

rappresenta il segnale nelle sue componenti di frequenza, eliminando le informazioni legate alla fase e concentrandosi sulla distribuzione delle ampiezze. Wiener la esprime come:

$$S(f) = \sum_{m=-\infty}^{\infty} \phi(m) e^{-i2\pi f m}$$

[1942b, p. 6, eq. 0.432]

Lo spettro consente di distinguere tra le frequenze dominate dal segnale utile e quelle occupate dal rumore, fornendo un quadro chiaro per separare i due campi. Questo approccio si rivela particolarmente importante nel contesto della stabilità delle serie temporali.

Wiener caratterizza ciò che significa per la stabilità l'esistenza di queste serie temporali, dimostrando che la conoscenza di $\phi(m)$ e $S(f)$ è fondamentale per costruire predizioni affidabili. La predizione si basa sulla minimizzazione dell'errore quadratico medio [*mean square prediction error*]:

$$E = \int_{-\infty}^{\infty} |x(t) - \hat{x}(t)|^2 dt$$

[1942b, p. 56, eq. 2.01]

Questa formulazione stabilisce le condizioni sotto cui le predizioni possono essere fatte, fornendo una stima ottimale del comportamento futuro della serie temporale basata esclusivamente sulle sue proprietà statistiche.

La mossa di Wiener e Bigelow fu quella di idealizzare il problema, e assumere come appartenenti a una classe speciale di funzioni con proprietà statistiche note (le funzioni con covarianza espresse nelle complesse equazioni dello *Yellow Peril*), sia il movimento dell'aereo

componenti sinusoidali. Originariamente concepita per risolvere problemi di conduzione del calore, ha trovato applicazioni fondamentali, anche grazie al lavoro di Wiener, nelle telecomunicazioni, in particolare nella modulazione, processo che permette la trasmissione simultanea di più messaggi su una singola linea separandoli tramite variazioni di frequenza o passo.

Norbert Wiener ampliò ed estese l'uso della trasformata di Fourier in contesti avanzati, come nell'Analisi Armonica Generalizzata in *Generalized Harmonic Analysis*, [Wiener, 1930] e nei Teoremi Tauberiani di *Tauberian Theorems*, [Wiener, 1932]. Per l'analisi approfondita dal punto di vista matematico, si rimanda alla raccolta completa della produzione scientifica wieneriana commentata da illustri matematici, curata da P.R Masani, del quale si rimanda a [1990], Masani (ed.), 1979-1985, in particolare, per questo tema, al Vol. II.

che gli errori di registrazione. Se questo è possibile considerarlo il “rumore” [*noise*], il primo costruisce il messaggio [*message*]⁷⁸.

Il lavoro svolto “sul trattamento statistico del controllo del fuoco contraereo portò infine a una visione statistica generale nell’ingegneria delle comunicazioni” È con malcelato orgoglio che Wiener afferma l’impatto pionieristico dell’analisi contenuta in *Yellow Peril* nel pensare alla comunicazione in termini probabilistici, l’approccio statistico alla teoria della comunicazione è ora accettato quasi ovunque. [Wiener, 1953b, p.255 trad. mia]⁷⁹.

Qui possiamo notare un ulteriore esempio della naturale torsione “neuromorfica” delle idee di Wiener – interessato a comprendere il funzionamento biologico del cervello. Racconta il matematico del M.I.T. come fosse interessato all’applicazione dei metodi di matematica probabilistica condensati nello *Yellow Peril* allo studio delle “onde cerebrali” [*brain waves*]. A questo scopo, Wiener racconta come non cessasse mai di “importunare i miei compagni neurofisiologi con la richiesta che prestassero un ascolto benevolo a questi metodi e che, se possibile, li sperimentassero su alcuni dati sperimentali” [Wiener, 1953b, p.288 trad. mia]⁸⁰.

Wiener era convinto che lo studio, attraverso l’elettroencefalogramma, delle “correnti elettriche vaganti nel cervello [...] avrebbe gettato nuova luce sulla fisiologia del cervello e sui fenomeni mentali a essa associati” [Wiener, 1953b, p.289 trad. mia]⁸¹ Le *brain waves*, secondo Wiener, “così come le osserviamo originariamente, sono una mescolanza di fenomeni molto diversi, simile a ciò che troveremmo, ad esempio, se osservassimo le correnti elettriche vaganti attorno a una macchina calcolatrice o a una macchina di controllo” [Wiener, 1953b, p. trad. mia]⁸². Notiamo come qui la metafora cibernetica di analogia tra cervello e calcolatore prenda

⁷⁸ Seppur matematicamente complesso, *Yellow Peril* rappresenta un lavoro “cruciale” [Galison, 1994, p.41 n.28 trad. it.] nello sviluppo della cibernetica, completando le fondamenta concettuali della nuova scienza. Questo contributo, insieme agli studi sulla natura strutturale dei sistemi organici e artificiali, fornisce all’analisi dei servomeccanismi di feedback le basi matematiche per lo studio della comunicazione e dei messaggi, poi sviluppate e pienamente dispiagate da Wiener nelle sue opere maggiori. Si comprende così perché la proposizione fondamentale della cibernetica sia definita come “segnale = messaggio + rumore” [“The basic proposition of cybernetics [is] that signal = message + noise”, Masani, 1990, p.19].

⁷⁹ “The work I did on the statistical treatment of anti-aircraft-fire control has led eventually to a general statistical point of view in communication engineering. The years that have gone by between then and now have secured the general acceptance of this point of view in communication engineering, but they have also led to much more”.

⁸⁰ “For years I had the intention of using these tools in every region in which they seemed apt. From the very beginning of the study of brain waves, ever since I had contact with some of the original electroencephalographers in Arturo’s seminar at the Harvard Medical School, I had felt that this field was one in which I could accomplish something; and I have never ceased to importune my neurophysiologist comrades with requests that they give a sympathetic hearing to these methods and that they try them out if possible on some experimental data”

⁸¹ “In the early days of brain-wave work, it was supposed that the stray electric currents in the brain, as observed through the scalp, would throw a new light on the physiology of the brain and the associated mental phenomena”

⁸² “The reason is that the brain waves as we see them originally are a mixture of very varied phenomena such as we would find if, for example, we observed the stray electric currents around a computing machine or a control machine”.

corpo sui fondamenti matematici *in nuce* della teoria dell'informazione. Il comportamento – *omesostatico*, regolato da *feedback* – di ogni ente poteva, secondo Wiener, essere studiato statisticamente.

Nella nascente tecnologia dell'elettroencefalogramma, infatti, “c'è molta informazione contenuta in questi tracciati a inchiostro, ma è come l'informazione che possedevamo sulla lingua egizia nei giorni precedenti alla Stele di Rosetta, la quale ci fornì la chiave per decifrare la scrittura egizia” [Wiener, 1953b, trad. mia]⁸³. Wiener voleva trovare, attraverso il suo studio della matematica statistica delle correnti, la chiave per decifrare questo codice.

La speranza di Wiener era che strumenti appena inventati come l'interferometro del fisico americano A.A. Michelson (1852–1931) e la matematica dell'autocorrelazione, con gli strumenti elaborati nello *Yellow Peril*, potessero fornire gli strumenti per decifrare la “Stele di Rosetta” delle onde elettriche cerebrali. Come egli stesso scrive: “Negli ultimi anni ho fatto parte di un gruppo, che comprende persone di vari laboratori del M.I.T. e del Massachusetts General Hospital, impegnato a cercare la Stele di Rosetta per la scrittura delle onde cerebrali mediante l'analisi armonica” [Wiener, 1953b, p.289 trad. mia]⁸⁴.

Da un'altra prospettiva, a questi problemi parallelamente lavorava un'altra figura pionieristica che ha contribuito con la sua opera a plasmare il mondo contemporaneo: Claude Shannon. Durante il suo ultimo anno all'Università del Michigan, dove studiò matematica ed ingegneria elettrotecnica, vide in una bacheca un annuncio con un'offerta di lavoro per uno studente laureato. Vannevar Bush, che di Shannon divenne il mentore, cercava un assistente per seguire il suo Analizzatore Differenziale, presso il MIT.

"L'anno prima dell'arrivo di Shannon in Massachusetts, la gargantuesca macchina analogica di Bush - presso la quale orde di professori, tecnici e studenti si recavano come 'supplici in pellegrinaggio' [Gleick, 2011, p.162 trad. it.] affinché resolvesse le loro equazioni differenziali - iniziava a mostrare i propri limiti: gli stessi limiti che, almeno dal punto di vista dell'hardware, avevano già condannato al fallimento le prospettive di Wiener di un utilizzo pratico del predittore antiaereo.

⁸³ “They speak a language of their own, but this language is not something that one can observe precisely with the naked eye, by There merely looking at the ink records of the electroencephalo- graph. is much information contained in these ink records, but it is like the information concerning the Egyptian language which we had in the days before the Rosetta Stone, which gave us the clue to the Egyptian script”.

⁸⁴ “Of recent years, I have been a member of a group involving people from the various laboratories at M.I.T. and from the Massachusetts General Hospital which has been endeavoring to of find the Rosetta Stone for the script of the brain waves by means of harmonic analysis.”

L'Analizzatore Differenziale, su cui torneremo più volte, non era una 'macchina unica', bensì un insieme di 'macchine in potenza', da riassemblare di volta in volta a seconda del singolo problema da affrontare, sacrificando l'efficienza sull'altare della versatilità. In questo modo incontrava uno dei primi colli di bottiglia della storia del calcolo attraverso macchine che finivano per compromettere la stessa ragione della loro esistenza, dato che lo scopo della macchina di Bush era compiere calcoli teoricamente eseguibili dal calcolatore umano con l'aiuto delle sole mente e mano" [Soni, Goodman, 2017, p.33 trad. mia]⁸⁵.

In risposta, Bush sognava un analizzatore che potesse essenzialmente ricostruirsi da solo al volo: "uno con controlli automatici che gli permettessero di passare da un'equazione all'altra senza pause, o persino di risolvere simultaneamente equazioni multiple interagenti" Una macchina, cioè, in cui gli interruttori prendessero il posto del lavoro dei cacciavite [Soni, Goodman, 2017, p.33 trad. mia]⁸⁶. All'interno dello spazio occupato dalle 100 tonnellate di componenti ronzanti e sferraglianti che costituivano l'Analizzatore, il lavoro di Shannon era riservato a "una piccola scatola con 100 interruttori che si aprivano e chiudevano, fissata all'analizzatore, un mondo dentro un mondo. Nella scatola c'erano i cervelli del cervello, gli interruttori e i relè che ora controllavano la macchina" [Soni, Goodman, 2017, p.34 trad. mia]⁸⁷.

Shannon, spese tre anni tra queste componenti della macchina di Bush, aprendo e chiudendo interruttori. Ma l'obiettivo del suo lavoro era diverso dallo studio e dall'impiego classico di questi dispositivi di commutazione, quello ad esempio impiegato nelle reti telegrafiche o telefoniche. L'obiettivo di Shannon era il "controllo". "Un centinaio di relè, interconnessi tra loro in modo complesso, attivandosi o disattivandosi in sequenze particolari, coordinavano l'Analizzatore differenziale". [Gleick, 2011, p.163 trad. it.].

In questo *micromondo* digitale, all'interno del mostro metallico analogico rappresentato dall'Analizzatore di Bush, Shannon, che cercava un argomento per la sua magistrale - memore di un corso di filosofia seguito all'università e dell'infanzia tra nella natia Gaylord, cittadina di

⁸⁵ "What Bush and his team had built was not so much a single machine as a huge array of machines in outline, to be rebuilt for every problem and broken down at every solution. It was versatility at the cost of efficiency; and because the analyzer's entire mission was to bring efficiency to computations that could at least in theory be worked out by human hands, these recurring bottlenecks compromised its whole reason for being."

⁸⁶ "In response, Bush dreamed of an analyzer that could essentially reassemble itself on the fly: one with automatic controls enabling it to turn from equation to equation without pausing, or even to solve multiple interacting equations simultaneously. Switches would take over the work of screwdrivers."

⁸⁷ "In the box were the brains of the brain, the switches and relays that now controlled the machine".

poche migliaia di anime nella campagna del Michigan, in cui da bambino si divertiva a costruire rudimentali telegrafi con il filo spinato che abbondava⁸⁸ – ebbe una intuizione.

All'interno di una macchina costruita per rendere automatici i calcoli, Shannon si ricordò dell'algebra e della logica Booleana⁸⁹, per cui “qualsiasi affermazione logica poteva essere catturata in simboli ed equazioni-e che queste equazioni potevano essere risolte con una serie di semplici regole, simili a quelle matematiche” [Soni, Goodman, 2017, p.35 trad. mia]⁹⁰, cioè, di un mezzo, per formalizzare il pensiero. Shannon capì per primo, che “la particolare notazione artificiale della logica simbolica, l’“algebra di Boole”, poteva essere utilizzata per descrivere i circuiti” [Gleick, 2011]

Shannon, che nel 1937 si era trasferito dal MIT ai Bell Labs per un lavoro estivo, iniziò a concretizzare le sue intuizioni. Lì si trovò immerso nella realtà della rete di circuiti più complessa e vasta del tempo, e il suo incarico consisteva nel contribuire a ottimizzarla, rendendola più efficiente e meno costosa attraverso strumenti matematici. Questo contesto si rivelò cruciale: fu proprio in quel periodo che Shannon, per la prima volta, “mise penna su carta, iniziando a intrecciare le connessioni che intravedeva tra l'analizzatore differenziale di Bush, la rete dei Bell Labs e la logica simbolica di Boole”⁹¹ [Soni, Goodman, 2017, p.38 trad. mia]⁹².

Il risultato fu la tesi completata alla fine del '37, sotto la supervisione del matematico del MIT Frank Hitchcock, *A Symbolic Analysis of Relay and Switching Circuits* [Shannon, 1939]

⁸⁸ Per la vicenda biografica e intellettuale di Shannon, e le relazioni del suo lavoro con quello di Boole si rimanda a Nahin, 2013; Gleick, 2011 e soprattutto a Soni, Goodman, 2017.

⁸⁹ George Boole (1815-1864) ha dato un contributo decisivo, in particolar modo nelle sue due opere fondamentali *The Mathematical Analysis of Logic* (1847) e *An Investigation of the Laws of Thought* [Boole, 1854], allo sviluppo della logica, probabilmente smuovendola dopo due millenni di torpore aristotelico [Davis, 2000, p. 60, trad. it.]. Egli dimostrò che la logica poteva essere trattata come un ramo dell'algebra, formalizzandola in un sistema simbolico. "La grande conquista di George Boole consistette nel dimostrare, una volta per tutte, che la deduzione logica poteva essere trattata come un ramo della matematica" [Davis, 2000, p. 60, trad. it.]. Il sistema logico e algebrico booleano, grazie all'intuizione di Shannon, ha avuto un impatto enorme sull'ingegneria elettrica e delle comunicazioni, permettendo di utilizzare in questo campo gli operatori algebrici AND, OR, NOT e IF. Per il rapporto tra il lavoro di Boole e quello di Shannon si veda Nahin, 2013. L'idea che i processi logici potessero essere formalizzati e ridotti a due valori fondamentali, 0 e 1, e che “l'algebra della logica è esattamente ciò che sarebbe l'algebra ordinaria se fosse limitata a due soli valori, 0 e 1” [Davis, 2000, p.51 trad. it.] ha tutt'ora una portata decisiva nella struttura logica dell'informatica, in cui un valore booleano rappresenta uno dei due stati logici possibili: vero (true) o falso (false). È fondamentale per operazioni logiche, strutture di controllo nei programmi e circuiti digitali. Per una panoramica completa sul ruolo di Boole nella storia della logica e del pensiero matematico, si rimanda a Davis (2000, pp. 38-69, trad. it.), Kneale, Kneale (1962, pp. 460-485, trad. it.) e Mangione, Rizzi (1993, pp. 96-170).

⁹⁰ “In Michigan, Shannon had learned (in a philosophy class, no less) that any statement of logic could be captured in symbols and equations-and that these equations could be solved with a series of simple, math-like rules. You might prove a statement true or false without ever understanding what it meant”

⁹¹ “Crucially, it was around this time that he first put pen to paper and began tying together the commonalities he sensed in Bush's analyzer, Bell's network, and Boole's logic”.

⁹² “Crucially, it was around this time that he first put pen to paper and began tying together the commonalities he sensed in Bush's analyzer, Bell's network, and Boole's logic.”

pubblicata lo stesso anno negli *Transactions of the American Institute of Electrical Engineers* e presentata in pubblico lo stesso anno a Washington D.C. e che il 1939, grazie all'interessamento di Bush, fece a vincere a Shannon l'Alfred Nobel Prize assegnato ogni anno dalle società ingegneristiche combinate degli Stati Uniti al miglior paper di ingegneria scritto da un autore sotto i 30 anni.

In quella che è stata definita in seguito "probabilmente la tesi di laurea magistrale più importante e anche la più famosa del secolo" [Gardner, 1985, p.166 trad. it.], Shannon compie un'operazione concettuale che nessuno aveva fatto prima di lui: far coincidere il mondo della logica e quello dell'elettricità, tra il calcolo proposizionale e quello dei circuiti.

La mossa di apertura di Shannon è perentoria. Dichiara infatti che "qualsiasi circuito è rappresentato da un insieme di equazioni, i cui termini corrispondono ai vari relè e interruttori del circuito" e che egli "sviluppa un calcolo per manipolare queste equazioni tramite semplici processi matematici, la maggior parte dei quali sono simili agli algoritmi dell'algebra ordinaria", calcolo che dimostrerà essere "esattamente analogo al calcolo delle proposizioni utilizzato nello studio simbolico della logica". Stabilita questa equivalenza, "il circuito può quindi essere immediatamente tracciato a partire dalle equazioni". E questo vale per i circuiti in serie, in parallelo e per circuiti che contengano qualunque tipo di connessione. [Shannon, 1939 trad. mia]⁹³.

Ciò di cui Shannon si rese conto era che, al di là della quantità di elettricità, da un relè all'altro si trasferisce uno *stato*: l'essere aperto o chiuso, spento o acceso, 0 e 1, i due stati fondamentali della logica algebrica di Boole: *vero* o *falso*. Posta l'impedenza [*hindrance*] di un circuito *a-b* "Il simbolo 0 (zero) sarà utilizzato per rappresentare l'impedenza di un circuito chiuso, mentre il simbolo 1 (unità) rappresenterà l'impedenza di un circuito aperto". Posti i due stati fondamentali, Shannon passa poi, nel suo asciutto stile matematico a fornire delle definizioni:

1. Il simbolo 0 (zero) rappresenta l'impedenza di un circuito chiuso, mentre il simbolo 1 (unità) rappresenta l'impedenza di un circuito aperto. Pertanto:

- Quando il circuito *a-b* è aperto : $X_{ab} = 1$

⁹³ "The method of attack on these problems may be described briefly as follows: any circuit is represented by a set of equations, the terms of the equations corresponding to the various relays and switches in the circuit. A calculus is developed for manipulating these equations by simple mathematical processes, most of which are similar to ordinary algebraic algorisms. This calculus is shown to be exactly analogous to the calculus of propositions used in the symbolic study of logic. The circuit may then be immediately drawn from the equations. By this method it is always possible to find the simplest circuit containing only series and parallel connections, and in some cases the simplest circuit containing any type of connection".

- Quando il circuito $a-b$ è chiuso: $X_{ab} = 0$

2. Due impedenze $X_{ab} = X_{cd}$, sono considerate uguali se:

- Quando $a-b$ è aperto, $c-d$ è aperto.

$X_{ab} = X_{cd}$, allora $c - d$ è aperto.

- Quando $a-b$ è chiuso, $c-d$ è chiuso.

$X_{ab} = X_{cd}$, allora $c - d$ è chiuso.

3. Connessione in serie (+): Il simbolo $+$ rappresenta la connessione in serie dei due circuiti terminali, le cui impedenze si sommano:

$$X_{totale} = X_{ab} + X_{cd}$$

4. Connessione in parallelo ($\cdot$): Il prodotto di due impedenze rappresenta la connessione in parallelo dei circuiti:

$$X_{totale} = X_{ab} \cdot X_{cd}$$

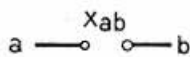


Fig.3

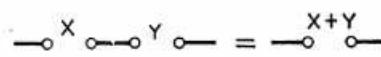


Fig.4

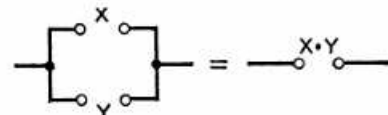


Fig.5

Fig. 3; 4; 5 In *A Symbolic Analysis of Relay and Switching Circuits*, Shannon, 1938.

Questa rappresentazione semplifica la manipolazione delle impedenze, *rendendola simile all'algebra numerica ordinaria* [Shannon, 1939, enfasi mia]⁹⁴.

⁹⁴ "The symbol 0 (zero) will be used to represent the hindrance of a closed circuit, and the symbol 1 (unity) to represent the hindrance of an open circuit. Thus, when the circuit $a-b$ is open, $X_{ab} = 1$ and when closed, $X_{ab} = 0$. Two hindrances, X_{ab} and X_{cd} , will be said to be equal if, whenever the circuit $a-b$ is open, the circuit $c-d$ is also open, and whenever $a-b$ is closed, $c-d$ is closed.

Now, let the symbol $+$ (plus) be defined to mean the series connection of the two terminal circuits whose hindrances are added together. Thus, $X_{ab} + X_{cd}$ is the hindrance of the circuit $a-d$ when b and c are connected together.

Un contatto o un interruttore a relè è rappresentato, come mostrano le figure 1, 2 e 3, attraverso simboli che associano lettere a specifiche funzioni di resistenza. La figura 1 mostra il simbolo del contatto o interruttore, con la lettera che indica la funzione di impedenza corrispondente. Nella figura 2 è illustrata l'interpretazione del segno "+" per l'addizione delle impedenze, mentre la figura 3 rappresenta il segno di moltiplicazione per le impedenze in parallelo [Shannon, 1939 trad. mia]⁹⁵.

Questa formalizzazione non solo semplifica il ragionamento sui circuiti, ma permette anche una traduzione diretta in termini logici e algebrici, rendendola particolarmente utile per applicazioni teoriche e pratiche.

L'insieme precedente di definizioni, per Shannon, rende validi i seguenti postulati:

1.a	$0 \cdot 0 = 0$	Un circuito chiuso in parallelo con un circuito chiuso è un circuito chiuso.
1.b	$1 + 1 = 1$	Un circuito aperto in serie con un circuito aperto è un circuito aperto.
2.a	$1 + 0 = 0 + 1 = 1$	Un circuito aperto in serie con un circuito chiuso, in qualsiasi ordine, è un circuito aperto.
2.b	$0 \cdot 1 = 1 \cdot 0 = 0$	Un circuito chiuso in parallelo con un circuito aperto è un circuito chiuso.
3.a	$0 + 0 = 0$	Un circuito chiuso in serie con un circuito chiuso è un circuito chiuso.
3.b	$1 \cdot 1 = 1$	Un circuito aperto in parallelo con un circuito aperto è un circuito aperto.
4	Ad ogni dato momento sia $X = 0$ o $X = 1$ ⁹⁶ :	

Similarly, the product of two hindrances, $X_{ab} \cdot X_{cd}$ or more $X_{ab}X_{cd}$ will be defined to mean the hindrance of the circuit formed by connecting the circuits a-b and c-d in parallel."

⁹⁵ "A relay contact or switch will be represented, as shown in figure 1, by the symbol associated with a letter indicating the corresponding hindrance function. Figure 2 shows the interpretation of the "+" sign as a series connection, while figure 3 represents the multiplication sign as a parallel connection. This choice of symbols makes the manipulation of hindrances very similar to ordinary numerical algebra.

⁹⁶

1.a	$0 \cdot 0 = 0$	A closed circuit in parallel with a closed circuit is a closed circuit.
1.b	$1 + 1 = 1$	An open circuit in series with an open circuit is an open circuit.

Armato di questa piattaforma logica, Shannon, dopo aver stabilito i teoremi che descrivono le regole algebriche per le connessioni in serie e parallelo nei circuiti a relè – con l’obiettivo di semplificare l’analisi logica dei circuiti attraverso la dualità tra addizione e moltiplicazione – è “ora in grado di dimostrare l’equivalenza di questo calcolo con alcune parti elementari del calcolo delle proposizioni”, utilizzando “l’algebra logica, originata da George Boole, come metodo simbolico per indagare le relazioni logiche” [Shannon, 1939 trad. mia.]⁹⁷.

Shannon ha incorporato nelle fisicità elettrica dei circuiti i concetti chiave dell’algebra logica di Boole: AND per le connessioni in serie (perché la corrente deve passare attraverso due interruttori successivamente e non raggiunge la sua destinazione se entrambi non le consentono il passaggio); OR per le connessioni in parallelo (perché la corrente può passare attraverso uno dei due interruttori, o entrambi); e NOT per rappresentare la negazione (perché lo stato del circuito viene invertito: aperto diventa chiuso, e viceversa).

Shannon può così proporre la sua equivalenza tra la logica proposizionale e l’analisi simbolica dei relè [Shannon, 1939, nella Table I. riprodotta in originale in nota]⁹⁸:

2.a	$1 + 0 = 0 + 1 = 1$	An open circuit in series with a closed circuit, in any order, is an open circuit.
2.b	$0 \cdot 1 = 1 \cdot 0 = 0$	A closed circuit in parallel with an open circuit is a closed circuit.
3.a	$0 + 0 = 0$	A closed circuit in series with a closed circuit is a closed circuit.
3.b	$1 \cdot 1 = 1$	A closed circuit in parallel with a closed circuit is a closed circuit.
4	At any given time either $X = 0$ or $X = 1$	

⁹⁷ “We are now in a position to demonstrate the equivalence of this calculus with certain elementary parts of the calculus of propositions. The algebra of logic, originated by George Boole, is a symbolic method of investigating logical relationships”

Table I. Analogue Between the Calculus of Propositions and the Symbolic Relay Analysis

Symbol	Interpretation in Relay Circuits	Interpretation in the Calculus of Propositions
X	The circuit X	The proposition X
0	The circuit is closed.....	The proposition is false
1	The circuit is open.....	The proposition is true
$X + Y$...	The series connection of circuits X Y	The proposition which is true if either X or Y is true
XY	The parallel connection of circuits X and Y	The proposition which is true if both X and Y are true
X'	The circuit which is open when X is closed,... and closed when X is open	The contradictory of proposition X
$=$	The circuits open and close simultaneously.....	Each proposition implies the other

Simbolo	Interpretazione nei circuiti a relè	Interpretazione nel calcolo delle proposizioni
X	Il circuito X	La proposizione X
0	Il circuito è chiuso	La proposizione è falsa
1	Il circuito è aperto	La proposizione è vera
$X + Y$	La connessione in serie dei circuiti X e Y	La proposizione che è vera se X o Y è vera
XY	La connessione in parallelo dei circuiti X e Y	La proposizione che è vera se sia X che Y sono vere
X'	Il circuito che è aperto quando X è chiuso e chiuso quando X è aperto	La contraddittoria della proposizione X
=	I circuiti si aprono e si chiudono simultaneamente	Ogni proposizione implica l'altra

Il risultato più importante, e più radicale, di Shannon emergeva quasi implicitamente, la capacità dei circuiti di trattare l'uguaglianza come se fosse un condizionale [Soni, Goodman, 2017, p.42]⁹⁹, cioè la possibilità di questi di compiere scelte del tipo “se...allora”, di implicazione [Gleick, 2011, p.164 trad. it] come mostrano ad esempio i postulati 1.b e 3.a.

Shannon infine nella sua tesi trova il modo di sfruttare la sua intuizione, analizzando le trasformazioni reti da stella a maglia [*star to mesh*] e da delta a Y [*delta to wye*]; serrature elettriche a cinque pulsanti [*electric combination lock*] con allarme. Egli aveva preso la lezione della logica appresa nei suoi studi universitari, quella, già citata, per cui “qualsiasi affermazione logica poteva essere catturata in simboli ed equazioni-e che queste equazioni potevano essere risolte con una serie di semplici regole, simili a quelle matematiche.” [Soni, Goodman, 2017]. Questo permetteva di dimostrare che una affermazione era vera o falsa senza disinteressandosi della comprensione del significato. “Anzi, era meno distrattivo non capirla: la deduzione poteva essere automatizzata” [Soni, Goodman, 2017, p.35]¹⁰⁰

⁹⁹ “The most radical result of Shannon’s thesis was largely implied, not stated, and its import only became clear with time. The implication gets clearer when we realize that Shannon, following Boole, treated the equal sign as a conditional: “if.”

¹⁰⁰ “You might prove a statement true or false without ever understanding what it meant. You would be less distracted, in fact, if you chose not to understand it: deduction could be automated”.

Shannon compie un passo ulteriore nella via segnata da Boole nella sua algebrizzazione della logica. Se il logico inglese aveva iniziato ad assegnare simboli come x o y a classi rappresentanti collezioni di oggetti nello stesso modo in cui lo si faceva nell'algebra, Shannon mostra che una volta *logicizzati*, gli stessi relè in cui aveva incorporato la logica non erano più imprescindibili. Il sistema avrebbe potuto funzionare in qualsiasi implementazione – così come l'intima struttura a retroazione negativa individuata da Wiener e dai suoi collaboratori era comune a qualunque sistema organico o inorganico – dotata di un sistema logico binario, la struttura base delle *leggi del pensiero* booleanae. L'idea che una funzione logica, comune a qualsiasi realizzazione fisica – relè o pensiero, macchina o cervello – potesse essere indipendente, e implementabile in ogni sostrato ha qui uno snodo genetico cruciale.

Questa logica dell'astrazione avrebbe compiuto un ulteriore passo nell'altro grande contributo di Shannon, che sarà analizzato a breve, *la teoria matematica della comunicazione*. Ma è importante notare come, teorizzando trasformando l'arte della progettazione e dei circuiti in una scienza rigorosa come la matematica, *A Symbolic Analysis* non aveva trovato solo un modo per ridurre undici passaggi ingegneristici a due. Mostrava che vi era una capacità decisionale, nella dimensione del “se...allora” sopra mostrata, immanente ai circuiti elettrici, che li rendeva in grado, potenzialmente, di risolvere problemi logici enormemente complessi, in modo più veloce e più affidabile di un deduttore o un calcolatore umano.

A seconda dell'input, gli stessi interruttori potevano dare risposte diverse. Facendo un salto antropomorfo-un circuito poteva decidere. Un circuito poteva fare logica. Molti circuiti potevano fare logiche enormemente complesse: potevano risolvere problemi logici e dedurre conclusioni da premesse con la stessa affidabilità “E poiché Boole aveva mostrato come ridurre la logica a una serie di decisioni binarie, vero-falso, qualsiasi sistema capace di rappresentare binari aveva accesso all'intero universo logico che aveva descritto. “*Le leggi del pensiero*” erano state estese al mondo inanimato” [Soni, Goodman, 2017, p.42 trad. mia enfasi mia]¹⁰¹.

L'automatizzazione del pensiero avanzata dall'algebrizzazione booleana aveva preso corpo nei circuiti in quanto capaci dell'operazione binaria basilare. Ma la materialità del relè o dell'interruttore non era importante in sé. McCulloch e Pitts in *A Logical Calculus of Ideas Immanent in Nervous Activity*, il 1943, pochi anni dopo la il lavoro di Shannon, avrebbero

¹⁰¹ “And because Boole had shown how to resolve logic into a series of binary, true-false decisions, any system capable of representing binaries has access to the entire logical universe he described. “The laws of thought” had been extended to the inanimate world”.

mostrato, come prenderemo avanti in esame, che questo avveniva anche per una speciale tipologia *organica* di circuiti, i neuroni del cervello.

A *Symbolic Analysis* nata dal lavoro di Shannon sulla macchina di Bush, contribuì a segnare concettualmente la fine delle macchine analogiche. Dall'interno del meccanismo di controllo di una macchina, L'Analizzatore di Bush, su cui "incombevano come fantasmi ancestrali le macchine di Charles Babbage" ne sopraggiunse la fine attraverso un sentiero che lo stesso Babbage¹⁰² aveva solo iniziato a seguire, sebbene Shannon fosse non ne fosse a conoscenza [Gleick, 2011, p.161 trad. it.]. "Nella tesi di master di un assistente di ricerca c'era l'essenza della rivoluzione informatica ancora di là da venire" [Gleick, 201, p.164]. Rivoluzione che avrà per driver di un altro concetto in cui il contributo di Shannon è stato decisivo, quello di *informazione*.

b. Geni algebrici, voci sintetizzate

Trasferitosi al MIT per conseguire il dottorato di ricerca, Shannon recepì il consiglio avuto dal il suo mentore Vannevar Bush, di abbandonare "la cella monastica dell'Analizzatore

¹⁰² Charles Babbage (1791–1871), matematico, filosofo e ingegnere britannico, è considerato un pioniere della scienza dei calcolatori grazie alle sue idee rivoluzionarie sul calcolo automatico. Professore alla prestigiosa cattedra lucasiana, già di Newton, di matematica presso Cambridge, Babbage progettò la Macchina Differenziale per calcolare automaticamente tabelle matematiche usando il metodo delle differenze finite [*method of finite differences*]. Sebbene mai completata durante la sua vita, una sua ricostruzione contemporanea del 1991 ha confermato la validità del progetto. Il suo lavoro più ambizioso, la Macchina Analitica (*Analytical Engine*), descritta nell'omonimo capitolo [*On the Analytical Engine*] della sua autobiografia del 1864 anticipò molte caratteristiche dei moderni computer, come un'unità aritmetica [*the mill*], una memoria centrale [*the store*], un meccanismo di controllo [*the control mechanism*] e un sistema di input/output basato su schede perforate, ispirate al telaio Jacquard. La macchina era concepita come il primo dispositivo *general-purpose*, capace di essere programmato per eseguire operazioni differenti. Tuttavia, difficoltà tecnologiche, economiche e la mancanza di sostegno adeguato impedirono a Babbage di completarla, portando molti contemporanei a considerare irrealizzabile un progetto così avanzato. Babbage nel 1826, pubblicò *On a Method of Expressing by Signs the Action of Machinery*, un metodo di notazione meccanica per rappresentare il funzionamento delle macchine, fondamentale per lo sviluppo della sua *Analytical Engine*, anticipando quello di Shannon del 1937. Babbage, 1826. Sulla vita e il pensiero di Babbage si vedano Babbage, 1984 Hyman, 1986; Gleick, 2011. Sull'evoluzione delle macchine di Babbage si rimanda a Bromley, 1987. Un'analisi puntuale della macchina di Babbage è in Goldstine, 1972, pp.10-27. Sul lavoro del matematico inglese in relazione agli sviluppi dell'Intelligenza Artificiale simbolica si veda Haugeland, 1985, pp.119-125 trad. it.. Di primissimo piano è anche la figura di Ada Lovelace, la prima di una lunga serie di donne nella *computer science* la cui importanza è stata spesso sottovalutata. Figlia del poeta Byron, il suo contributo fu cruciale per comprendere il potenziale della Macchina Analitica. Una sua osservazione sulla presunta incapacità delle macchine di andare oltre ciò per cui sono state programmate è stata presa in considerazione da Alan Turing nel suo seminale lavoro del 1950, *Computer Machinery and Intelligence*. In esso, Turing analizza questa come una delle principali obiezioni all'attribuzione di intelligenza alle macchine, Cfr. Turing, 1950, pp.143-145 trad.it. 1994. Per il "software" scritto da Ada Lovelace per l'*Analytical Engine*, si veda Menabrea, 1842 (Lovelace tradusse l'articolo dal francese e lo ampliò con procedure per calcolare numeri di Bernoulli). Le lettere e le note scritte da Ada Lovelace sono raccolte in Toole, 1992. Per approfondimenti sulla sua vita e il suo pensiero, si rimanda a Chiaverini, 2017, mentre per il suo rapporto con Babbage si consulti Huskey H. D. e Huskey V. R., 1980.

differenziale” [Soni, Goodman, 2017, p.49 trad. mia]¹⁰³ e di applicare, per la dissertazione dottorale, la sua comprovata abilità nell'algebra simbolica alla genetica Mendeliana¹⁰⁴ [Kline, 2015, p.27].

Shannon si recò così a Cold Spring Harbor, sede di uno dei maggiori laboratori di genetica d’America e, al tempo stesso, di una delle più grandi vergogne degli Stati Uniti negli anni tra le due guerre mondiali: l’*Eugenics Record Office*¹⁰⁵. Questo, malgrado le proprie finalità aberranti, possedeva un *data set* di prim’ordine, pronto per essere sottoposto al trattamento matematico di Shannon, il cui lavoro in questo ambito culminerà, nel 1940, nella tesi *An algebra for theoretical genetics* [Shannon, 1940]. L’applicazione dei metodi statistici alla genetica richiedeva infatti nuovi strumenti per realizzare la sintesi tra le scoperte mendeliane e la selezione naturale darwiniana. “La domanda di naturalisti e retini per farfalle era crollata; la biologia, come la costruzione di computer, richiedeva matematici” [Soni, Goodman, 2017, p. 52, trad. mia]¹⁰⁶

Shannon parte postulando la realtà dei geni¹⁰⁷ [Shannon, 1940, p.892], i quali – più di un decennio prima che l’immagine della doppia elica del DNA emergesse grazie alla macchina a raggi X – potevano essere utilizzati soltanto come costrutto teorico. Egli escogita, come aveva già fatto per i relè e gli interruttori, un metodo di rappresentazione simbolica per poter operare algebricamente sulla dinamica della popolazione mendeliana [Shannon, 1940, p. 892]¹⁰⁸.

L’idea di fondo è la stessa che aveva ispirato il suo lavoro sui circuiti: la loro costruzione esatta era più frutto di un’arte, dell’abilità contingente dell’ingegnere o dell’elettrotecnico, che di una scienza propriamente detta. La mancanza di scientificità derivava da una rappresentazione inadeguata, incapace di consentire operazioni rigorose.

¹⁰³ “And so Shannon was to leave the monastic cell of the differential analyzer (filled, like a monastery, with shifts of men keeping quiet watch at all hours) and the even smaller cell of the circuit box”

¹⁰⁴ “Bush suggested that he apply his proven ability in symbolic algebra to Mendelian genetics for his dissertation”.

¹⁰⁵ “Traveling to Cold Spring Harbor in the summer of 1939, Shannon arrived at one of the greatest genetics laboratories in America, and one of the greatest scientific embarrassments: the Eugenics Record Office. It housed a treasury of genetic data, which Shannon drew on for his dissertation on theoretical genetics”. Nel 1910, quando la sede del movimento eugenetico americano aprì le sue porte, era considerato all’avanguardia da alcuni sostenitori dell’idea di selezionare le “famiglie più idonee” e sterilizzare le classi povere. Il fondatore del movimento sosteneva che il 3-4% della popolazione fosse un freno per la civiltà. Crollata la popolarità dell’eugenetica con l’ascesa al potere del nazismo. nel 1939, Vannevar Bush chiuse l’ufficio. [Soni, Goodman, 2017, pp.50-51].

¹⁰⁶ “Demand for naturalists and butterfly nets had cratered; biology, like computer building, demanded mathematicians”.

¹⁰⁷ “Hereafter we shall speak therefore as though the genes actually exist and as though our simple representation of hereditary phenomena were really true, since so far as we are concerned, this might just as well be so”.

¹⁰⁸ “In this paper an attempt will be made to develop an algebra especially suited to problems in the dynamics of Mendelian populations”.

Shannon ripropose lo stesso schema con i geni, elaborando un sistema simbolico per rappresentare geneticamente un individuo, mediante lettere e numeri, così da costruire “formule genetiche” [Shannon, 1940, p. 912, trad. mia]¹⁰⁹.

Ad esempio, un organismo con due coppie di cromosomi e quattro posizioni geniche poteva essere descritto con:

$$A_1 B_2 C_3 D_4 \quad E_4 F_1 G_6 H_1$$

$$A_3 B_1 C_4 D_3 \quad E_4 F_2 G_6 H_2$$

Qui, ogni riga rappresentava un cromosoma e ogni colonna una posizione genica, mentre i numeri in pedice indicavano gli alleli specifici per un tratto. Le interazioni tra gli alleli di madre e padre (es. A_1 e A_3) determinavano, nella formulazione di Shannon, le qualità ereditate dalla prole.

Shannon ampliò questo modello per rappresentare intere popolazioni con un simbolo algebrico, λ , “una notazione adeguatamente compatta e suggestiva” [Shannon, 1940 p.895 trad. mia]¹¹⁰, capace di sintetizzare tutti i geni rilevanti e le loro frequenze. Il simbolo λ permetteva di calcolare variazioni genetiche e simulare cambiamenti nel tempo, fornendo un metodo matematico per studiare l’evoluzione e la ricombinazione genetica su scala macroscopica, er come nel caso dei relè - “simulare il futuro su carta” [Soni, Goodman, 2017, p.57 trad. mia]¹¹¹.

Il metodo e la notazione di Shannon permettevano di rispondere, con risorse non infinite di calcolo, a domande intorno al cambiamento genetico e alle caratteristiche di una data popolazione, individuando, in diversi intervalli di tempo o dopo diverse generazioni, i “fattori letali” [Shannon,1940, p.915 trad. mia]¹¹² o se avvenissero incroci causali con altri gruppi. Questo era possibile, nel lavoro di Shannon, mediante strumenti come l’equazione “mostro” “di dodici righe di parentesi e esponenti interconnessi-che calcolava le frequenze di tre diversi

¹⁰⁹ “Genetic formulae”.

¹¹⁰ “A suitably compact and suggestive notation”. Shannon si riferisce al potere di un’espressione di una notazione di essere “compatta” nel senso di poter essere usata come variabile in una equazione, e “suggestiva” come possibilità di poter scomporre gli elementi individuali ad ogni ciclo per indagare i risultati.

¹¹¹ “As before, a wise choice of symbols-addition for a parallel circuit, or a grid of variables for chromosomes-would allow Shannon to simplify and simulate the future on paper. The rest of his dissertation was a set of genetic theorems that put his algebraic tools to work.”

¹¹² “Lethal Factors”

alleli in qualsiasi popolazione dopo un certo numero di generazioni” [Soni, Goodman, 2017, p.57 trad. mia]¹¹³

Il lavoro di genetica algebrica di Shannon, esempio pionieristico di sistematizzazione matematica di enormi quantità di dati biologici, tuttavia non ebbe successo. Al contrario di *A Symbolic Analysis*, pur rimanendo un mirabile sforzo intellettuale che segnò la strada che avrebbe preso la biologia, *An Algebra for Theoretical Genetics* non ebbe un successo pratico tra i genetisti. Questo perché Shannon partiva da un livello d’astrazione troppo alto e dall’assunzione di una disponibilità completa della conoscenza dei dati genetici. “La mia teoria riguarda ciò che accade quando si possiedono tutti i fatti genetici. Ma le persone non li conoscono tutti, soprattutto per quanto riguarda gli esseri umani. Sono piuttosto esperte sul moscerino della frutta!” [Shannon, 1979, p.xxvii trad. mia]¹¹⁴.

Vedremo più avanti, invece, come un altro tentativo pionieristico simile, sviluppato da von Neumann, abbia aperto la strada a sviluppi come l’*Artificial Life* e ai prodromi dell’attuale Intelligenza Artificiale “concessionista”.

Shannon perse interesse nella sua tesi di dottorato che non fu mai pubblicata prima del 1993 con l’uscita dei *Collected Papers* dello matematico statunitense.

Shannon, intanto, mentre si divertiva tra i genetisti [Soni, Goodman, 2017, p.59]¹¹⁵ aveva già in mente un altro progetto. Come scrisse in una lettera al suo mentore Vannevar Bush:

¹¹³ “Most important, he derived the equation-a twelve-line monster of interlocking brackets and exponents-that gave the frequencies of three different alleles in any population after any number of generations”. L’equazione in questione, qui riportata, è quella elaborata relativamente al Teorema III - che descrive come rappresentare matematicamente le frazioni di una popolazione genetica in base alle combinazioni di *loci* genici, permettendo calcoli su ricombinazioni e tre frequenze alleliche- da Shannon [1993, pp.899-902]. Rispetto all’equazione (11) del Teorema III [Shannon, 1940, p.901], l’equazione sotto riportata mostra la capacità della variabile λ elaborata da Shannon di rispettare i canoni di compattezza e suggestività necessari in matematica per poter operare su simboli sistematizzando, semplificando e rendendo rigorosa la materia – siano essi *biodata* o circuiti – al livello di astrazione più basso a cui si riferiscono.

$$v_{j_1 j_2 \dots j_m}^{i_1 i_2 \dots i_n} = \lambda_{j_1 j_2 \dots j_m}^{i_1 i_2 \dots i_n} \times \mu_{j_1 j_2 \dots j_m}^{i_1 i_2 \dots i_n} = \frac{1}{2} [p_{00\dots 0} \lambda_{00\dots 0}^{i_1 \dots i_n} + p_{10\dots 0} \lambda_{1\dots 0}^{i_1 \dots i_n} + \dots + p_{11\dots 1} \lambda_{1\dots 1}^{i_1 \dots i_n}]$$

¹¹⁴ My theory has to do with what happens when you have all the genetic facts. But people don't know all of them, especially for humans. They are pretty well versed on the fruitfly!. Shannon si rese ovviamente conto della difficoltà di compiere esperimenti sugli umani per ottenere i dati necessari. La disponibilità dei dati dell’*Eugenics Record Office* era una eccezione dovuta all’aberrazione morale che animava l’ente.

¹¹⁵ “Shannon [...] “I had a good time acting as a geneticist for a couple years” he later joked”

Di tanto in tanto ho lavorato a un'analisi di alcune delle proprietà fondamentali dei sistemi generali per la trasmissione di informazioni, inclusi la telefonia, la radio, la televisione, il telegrafo, ecc. Praticamente tutti i sistemi di comunicazione possono essere ricondotti alla seguente forma:

$f_1(t) \rightarrow \boxed{T} \rightarrow F(t) \rightarrow \boxed{R} \rightarrow f_2(t)$ [Shannon 16 febbraio 1939, lettera a Bush, p.455]¹¹⁶.

Era con queste idee in mente, con la benevolenza di Bush e un PhD in tasca, che Shannon entrò a Princeton per una *fellowship* post-doc presso lo IAS (*Institute for Advanced Study*)¹¹⁷ l'istituto che ospitava titani della matematica e della fisica come, tra gli altri, Kurt Gödel (1906-1978), Albert Einstein (1879-1955) e John von Neumann. Qui studiò a guida di Hermann Weyl (1885-1955), successore di Hilbert alla cattedra di Gottinga, che tuttavia aveva ormai abbandonato il formalismo del maestro per abbracciare posizioni intuizioniste [Davis, 2000, pp. 109-135, trad. it.]. Weyl era giunto a Princeton nel 1933, dopo la presa del potere di Hitler, nonostante la sua "arianità". Fu in questo contesto che Shannon conobbe Warren Weaver ed entrò in contatto con gli ambienti dell'NDRC, i quali lo convinsero a posticipare la sua borsa di studio per lavorare a progetti di carattere applicato, legati – come nel caso di Wiener – alla progettazione e alla teoria predittiva del controllo di tiro [Kline, 2015, pp. 29-39].

Shannon ebbe in questo periodo problemi di carattere personale¹¹⁸. Uscì da questo momento buio accettando l'offerta del matematico Thornton Fry (1892-1991) per tornare

¹¹⁶ "Dear Dr. Bush, Off and on I have been working on an analysis of some of the fundamental properties of general systems for the transmission of intelligence, including telephony, radio, television, telegraphy, etc. Practically all systems of communication may be thrown into the following form: "

$$f_1(t) \rightarrow \boxed{T} \rightarrow F(t) \rightarrow \boxed{R} \rightarrow f_2(t)$$

$f_1(t)$ rappresenta una funzione generale del tempo che descrive l'informazione [*intelligence* nell'originale, non ancora *information*] da trasmettere (ad esempio, la pressione sonora in radio/telefonia o la tensione in uscita di un iconoscopio in televisione). T è l'elemento di trasmissione. Opera su $f_1(t)$ attraverso processi come modulazione e distorsione, trasformandola in una nuova funzione $F(t)$ che viene effettivamente trasmessa. $F(t)$ È la funzione di tempo trasmessa, ad esempio l'onda elettromagnetica emessa da un trasmettitore. Non è necessariamente simile a $f_1(t)$ ma è correlata. R è l'elemento ricevente. Trasforma $F(t)$ in una terza funzione $f_2(t)$ che idealmente dovrebbe essere il più simile possibile a $f_1(t)$. $f_2(t)$ è il segnale ricevuto dopo la trasformazione da parte di R. In un sistema ideale, sarebbe una replica esatta di $f_1(t)$ ma nella realtà ci sono sempre distorsioni..[Ibidem].

¹¹⁷ Per una storia dello IAS particolarmente significativa ai fini di questo lavoro, in quanto intrecciata con quella lo sviluppo dei calcolatori digitali e con le vicende umane e scientifiche dei protagonisti di questa rivoluzione di rimanda a Dyson, 2012.

¹¹⁸ Norma Barzman, prima moglie di Claude Shannon, lasciò il matematico nel 1941, in quanto lo credeva depresso e non disposto a farsi aiutare. Trasferitasi a Hollywood, divenne una sceneggiatrice di successo, autrice di *The Locket* (1946) e collaboratrice del marito Ben Barzman in film celebri come *El Cid* (1961) con Charlton Heston e Sophia Loren, di cui divenne amica, e *The Blue Max* (1966) con George Peppard. Esiliata in Francia negli anni '50

stabilmente ai Bell Labs, dei quali dirigeva il dipartimento di matematica, impegnati in prima fila nella ricerca bellica.

Non è chiaro ancora l'intera portata del lavoro di Shannon nel campo militare ai Bell Labs, essendo alcune delle sue attività tutt'oggi, dopo più di ottant'anni ancora segrete, ma si può affermare con ragionevole sicurezza che si occupò di tre ambiti: direttori antiaerei; sistemi di comunicazione segreta e crittografia [Kline, 2017, p.30 trad. mia]¹¹⁹. Tutti decisivi, assieme alle sue idee sulla matematizzazione algebrica, nel plasmare il contributo di Shannon alla teoria dell'informazione.

Per quanto il primo progetto, Shannon collaborò indirettamente con Wiener, rimanendo influenzato dall'elaborazione della matematica statistica esposta poi nello *Yellow Peril*. Shannon partecipò anche ad un incontro con l'inventore di computer John Atanasoff (1903-1995)¹²⁰, a cui erano presenti Stiglitz e Weaver, gli stessi che nel luglio del 1942 furono "perturbati" dall'apparecchio predittore di Wiener¹²¹.

Il secondo settore di lavoro di Shannon fu altrettanto decisivo. Collaborò al Progetto X, noto come SIGSALY (o anche come Ciphony I o Green Hornet), dal budget milionario [Burke, 2002, p.210]¹²², un sistema di comunicazione segreto sviluppato per garantire conversazioni sicure tra Franklin Roosevelt e Winston Churchill durante la Seconda Guerra Mondiale. Declassificato nel 1975, SIGSALY fu testato per la prima volta sull'Atlantico nell'autunno del

in quanto inserita lista nera del Comitato per le Attività Antiamericane, a causa della sua militanza per il Partito Comunista. La sua vicenda è narrata nella sua autobiografia, che contiene anche una foto inedita di Shannon del 1939, *The Red and the Blacklist: The Intimate Memoir of a Hollywood Expatriate* [Barzman, 2003],

¹¹⁹ "Although the full extent of Shannon's activities during the war is unknown because some of it is still classified, we know that he worked in three areas: antiaircraft directors, a secret communications system, and the theory of cryptography"

¹²⁰ John Atanasoff (1903-1995), fisico e professore all'Iowa State College, sviluppò tra il 1938 e il 1942 uno step cruciale nella storia della computazione con l'invenzione dell'Atanasoff-Berry Computer (ABC), considerato il primo prototipo di computer elettronico digitale. Secondo lo storico Paul Ceruzzi, fu il primo a utilizzare tubi a vuoto per eseguire calcoli matematici ad alta velocità, un'innovazione fondamentale rispetto ai calcolatori meccanici. I tubi a vuoto, come vedremo in dettaglio, sono dispositivi elettronici che controllano il flusso di corrente in un circuito, permisero di superare i limiti di velocità e complessità delle macchine precedenti. Progettato per risolvere sistemi di equazioni lineari, l'ABC introdusse anche l'uso del sistema binario per semplificare i circuiti di calcolo, anticipando i principi base dei moderni computer digitali. Ceruzzi, 2015, pp.33-35. Sul lavoro di Atanasoff, che avrà conseguenze nello sviluppo dell'ENIAC, si veda Burks, Burks, 1988.

¹²¹ Un cannone contraereo era già un calcolatore analogico: doveva convertire quelle che erano, a tutti gli effetti, equazioni differenziali del secondo ordine in movimenti meccanici", elaborando dati da radar e telemetri, e filtrando il rumore per compensare errori, in una riconoscibile analogia con il rumore delle linee telefoniche di cui i massimi esperti erano proprio ai Bell Labs e alla cui matematizzazione l'opera di Wiener contribuì in modo decisivo. Il problema dei cannoni era trattato un "caso speciale di trasmissione, manipolazione e utilizzazione di informazione" [Gleick, 2011, p.175 trad. it]

¹²² Per la storia della crittografia si rimanda a Kahn, 1967. Per le vicende dei tentativi di automatizzare la crittanalisi invece si veda il lavoro, secretato dalla National Security Agency (NSA) fino al 2013, di Burke, 2002.

1942. La tecnologia rivoluzionaria al centro del progetto era il *Vocoder*, inventato da Homer Dudley (1986-1980), ingegnere dei Bell Labs, il 1938.

Dudley ipotizzò che, poiché i suoni della voce umana, a un livello basilare, non erano altro che vibrazioni nell'aria, essi potessero essere riprodotti meccanicamente. La voce umana, dunque, poteva essere generata da una macchina. Partendo dalla neurofisiologia della parola, considerò la "natura vettoriale del linguaggio" [Dudley, 1940]¹²³, e interpretò il meccanismo discorsivo come un circuito. Da queste premesse nacquero due dispositivi: uno per codificare elettronicamente il parlato, il *Vocoder* (abbreviazione di *Voice Encoder*) – che dopo la guerra trovò numerosi impieghi, ad esempio nella musica pop – e uno per invertire il processo e generare parlato meccanico, il *Voder* (*Voice Operation Demonstrator*). Il *Vocoder* scomponiva il parlato nelle sue componenti fondamentali, come tono e timbro, trasmettendo solo le informazioni essenziali; il *Voder*, invece, era una consolle manovrata da un operatore che ricostruiva il parlato a partire da quei dati sintetizzati.

Nella riduzione meccanica del messaggio, vi era una sorta di eliminazione della ridondanza del discorso umano. La compressione operata dal *Vocoder* permetteva di trasmettere meno dati ma mantenere la comprensibilità del messaggio, riducendo il rischio di intercettazione nemica. "Il sistema campionava il segnale vocale cinquanta volte al secondo (lo "quantizzava" o "digitalizzava") e poi lo mascherava applicando una chiave casuale, che assomigliava molto al rumore circuitale ben noto agli ingegneri" [Gleick, 2011, p.189 trad. it.]

Una delle sfide della secretazione della comunicazione è che "ogni nuova lettera o parola introdotta in un messaggio apre una nuova possibilità di essere individuata dal nemico; meno si comunica, in altre parole, meglio è" [Soni, Goodman, 2017, p.100 trad. mia]¹²⁴. Il *Vocoder* riproduceva solo il necessario, limitando la parte di comunicazione esposta alla possibile decrittazione nemica.

La macchina di Dudley, inizialmente concepito per usi civili, fu poi integrato nel sistema SIGSALY¹²⁵, che utilizzava chiavi crittografiche casuali per garantire comunicazioni sicure. Shannon, fu uno dei 30 membri del team che lavorava su questo progetto.

¹²³ "Carrier nature of speech".

¹²⁴ "One of the central challenges of a code-writing system is this: each new letter or word introduced into a message opens up a fresh possibility of detection by the enemy; the less that's communicated, in other words, the better".

¹²⁵ Su SIGSALY, anche da un punto di vista ingegneristico e tecnologico, si veda Boone, Peterson, 2000 lavoro del *Center for Cryptologic History* dell'NSA, l'agenzia per la sicurezza nazionale statunitense.

SIGSALY non solo permise comunicazioni sicure durante la guerra, ma rappresentò una pietra miliare nell'applicazione pratica della compressione e crittografia vocale. Alla fine della guerra, il progetto fu premiato per la sua straordinaria innovazione. Tuttavia, la segretezza era tale che A.B. Clark rappresentante dei Bell Labs accettò il premio con una telefonata criptata: "Phrt fdygui jfsowria meeqm wuiosn jxolwps fuekswusjnvkci! Grazie!". [Sterling, 2010-11].

Il terzo settore del lavoro di Shannon ai Bell Labs, la crittografia, era strettamente legato agli altri due. Come ricostruisce mirabilmente Andrew Hodges nella sua biografia di Alan Turing, Shannon incontrò più volte il matematico inglese, di cui rappresentava il "vero omologo [...] in quanto a profondità intellettuale" [Hodges, 1983, p.327 trad. it.]¹²⁶, impegnato in prima linea nella battaglia per la decrittazione dei messaggi nazisti, cifrati dalla macchina Enigma¹²⁷.

¹²⁶ Alan Turing arrivò negli Stati Uniti per la seconda volta il 13 novembre 1942, dopo un suo primo soggiorno a Princeton. Dopo un periodo di missione presso gli omologhi americani del servizio di controspionaggio della marina dello CSAW (*Communication Supplementary Activities - Washington*), il 9 gennaio 1943 Turing giunse ai Bell Labs. Qui, nella caffetteria dei laboratori, durante l'ora del tè, incontrò più volte Claude Shannon, una figura che incarnava quello che si potrebbe definire un "accademico-filosofo-ingegnere," un ruolo che il sistema inglese non permetteva di esistere [Hodges, 1983, pp. 317-320, trad. it.]. Il clima di reciproca diffidenza che caratterizzava le istituzioni belliche e i servizi di controspionaggio, unito alla stretta segretezza, che permeava le relazioni tra i due lati dell'Atlantico, si rifletteva anche nel rapporto tra Shannon e Turing. Questo è testimoniato dalla reticenza di Shannon nel rispondere, anche a distanza di anni, a domande sugli incontri con Turing [Price, 1982]. Malgrado queste limitazioni, i due svilupparono un rapporto caratterizzato da grande stima reciproca, che emerge chiaramente dalla loro capacità di cogliere la profondità dei rispettivi lavori. Come evidenziato, vi era una "reciprocità" tra il loro pensiero e le loro intuizioni, che entrambi seppero riconoscere e apprezzare [Hodges, 1983, p. 328, trad. it.]. La visita di Shannon al laboratorio di Turing a Manchester nel 1950 fu l'ultima occasione in cui i due parlarono. Durante quell'incontro, discussero di computer, intelligenza artificiale e della possibilità di programmare una macchina per giocare a scacchi, un tema che affascinava entrambi. Turing condivise con Shannon alcune delle sue intuizioni tecniche, come l'idea di utilizzare segnali acustici per monitorare i processi all'interno di un computer, un esempio della sua creatività tecnica e della sua capacità di pensare fuori dagli schemi. Quattro anni dopo, nel 1954, Turing morì in circostanze tragiche a causa di un avvelenamento da cianuro. La sua morte fu ufficialmente classificata come suicidio, un atto che seguì alla sua condanna per "indecenza grave" (*gross indecency*), un'accusa mossa contro di lui per la sua omosessualità, allora criminalizzata nel Regno Unito.

¹²⁷ La macchina Enigma, progettata da Arthur Scherbius nel 1923, era originariamente una macchina per la cifratura destinata ad applicazioni commerciali, ma trovò il suo uso principale nell'ambito militare durante la Seconda Guerra Mondiale. Il dispositivo somigliava a una macchina da scrivere portatile, dotata di una tastiera e un pannello luminoso. Il cuore del sistema era rappresentato dai rotori intercambiabili (*Walzen*), che alteravano continuamente le connessioni elettriche interne. A ogni pressione di un tasto, almeno uno dei rotori ruotava, generando nuove configurazioni. Inoltre, l'introduzione del *Steckerbrett* (*plugboard*) permetteva ulteriori scambi di segnali tra le lettere, portando le combinazioni possibili a miliardi di miliardi. Questo rendeva la macchina apparentemente indecifrabile, dato che per decodificare un messaggio era necessario conoscere esattamente la configurazione giornaliera di rotori, posizioni iniziali (*Ringstellung*) e connessioni dello *Steckerbrett*. Quando nel 1939 Alan Turing entrò a Bletchley Park-un centro segreto britannico per la decrittazione-cominciò a lavorare su un problema centrale: decifrare i codici Enigma utilizzati dai tedeschi, in particolare nella guerra sottomarina degli U-Boot. Una macchina decifratrice era stata sviluppata da Marian Rejewski, del *Biuro Szyfrów* (Ufficio dei codici) polacco, chiamata così perché l'inventore ne ebbe l'idea mentre mangiava un dolce semifreddo, chiamato in francese la *bombe*. La Bomba polacca si basava su una vulnerabilità specifica nelle procedure operative tedesche. I tedeschi, infatti, ripetevano il messaggio iniziale due volte, usando la stessa configurazione dei rotori. Questo permetteva ai polacchi di ricavare informazioni essenziali sul cablaggio interno dei rotori e sulle impostazioni giornaliere di Enigma. La macchina di Rejewski testava rapidamente tutte le possibili configurazioni dei rotori e delle posizioni iniziali per individuare la chiave corretta. Tuttavia, nel 1938 i tedeschi introdussero nuovi rotori, aumentarono le combinazioni possibili e modificarono le procedure operative, eliminando la ripetizione del messaggio iniziale. Questi cambiamenti resero la Bomba polacca obsoleta. Turing progettò una nuova macchina

Shannon, appassionato di crittografia sin dall'infanzia¹²⁸, aveva lavorato su SIGSALY fu incaricato di dimostrare matematicamente la sicurezza del sistema, applicando la teoria delle probabilità per verificare l'inviolabilità degli algoritmi crittografici. Aveva lavorato sulla dimostrazione dell'inviolabilità del sistema Vernam, un metodo meccanizzato di crittografia inventato durante la Prima Guerra Mondiale per inviare messaggi segreti tramite linee telegrafiche. Nel 1945, Shannon ricompose i diversi fili intellettuali maturati durante la guerra, dalle indagini sui sistemi di controllo del fuoco al lavoro sul Progetto X, con la stesura nel 1945 di *A Mathematical Theory of Cryptography* (secretato fino al 1949) [Shannon, 1945].

Nel lavoro del 1945, probabilmente influenzato dai dialoghi con Turing, emergono profonde affinità con le idee che Shannon avrebbe sviluppato nella sua opera maggiore del 1948, *A Mathematical Theory of Communication*. Lo stesso Shannon ne dà testimonianza:

Durante la Seconda Guerra Mondiale i Bell Labs stavano lavorando su sistemi di segretezza. Avevo lavorato su sistemi di comunicazione e fui assegnato a alcuni dei comitati che studiavano le tecniche crittoanalitiche. Il lavoro sia sulla teoria matematica delle comunicazioni che sulla crittologia andò avanti contemporaneamente a partire dal 1941. Ho lavorato su entrambi insieme e ho avuto alcune

Bomba, completata nella primavera del 1940 e soprannominata inizialmente "Vittoria". Rispetto alla versione polacca, questa era un miglioramento radicale, sia fisicamente che tecnicamente. La Bomba britannica era alta circa due metri, larga oltre novanta centimetri e pesante più di una tonnellata. Costruita con 36 tamburi rotanti che replicavano i rotori dell'Enigma, conteneva circa 15 chilometri di cablaggi e oltre un milione di connessioni saldate. Il costo per una singola unità era di circa 650.000 sterline, una somma equivalente al costo di un bombardiere Lancaster. La macchina si basava sui *cribs*, frammenti prevedibili di testo presenti nei messaggi tedeschi, come formule meteorologiche o frasi standard. Ad esempio, termini come "*WETTER FUER DIE NACHT*" (tempo per la notte) erano utilizzati frequentemente dai meteorologi tedeschi. Questi *cribs* permettevano di identificare possibili configurazioni iniziali dei rotori. La Bomba testava milioni di combinazioni al secondo, cercando di allineare i tamburi in modo tale da riprodurre il messaggio originale. Una volta trovata la configurazione corretta, era possibile decifrare l'intero messaggio con una macchina Enigma replica. La Bomba di Turing era progettata per essere altamente adattabile ai cambiamenti delle procedure tedesche e molto più veloce ed efficace rispetto alla versione polacca. In particolare, il suo utilizzo del *plugboard* (*Steckerbrett*) rappresentava un passo avanti fondamentale. La macchina polacca, infatti, non era in grado di gestire le complessità aggiuntive introdotte dal *plugboard*, che aumentava esponenzialmente il numero di configurazioni possibili. Grazie alla Bomba di Turing, Bletchley Park divenne una vera "fabbrica di decifrazione", arrivando a decifrare oltre 84.000 messaggi al mese nel 1943. Questo lavoro, secondo alcune stime, abbreviò la durata della guerra di almeno due anni, salvando tra i 14 e i 21 milioni di vite umane. Il primo messaggio fu decifrato dalla Bomba nella primavera del 1940. Su Enigma e gli sforzi di Turing a Bletchley Park – e delle conseguenze scientifiche del suo lavoro – e dei servizi controspionaggio per decrittare si veda, Copeland, 2012, pp.49-159 trad. it.; Hodges, 1983, pp.212-408 trad. it. il funzionamento tecnico di Enigma si trova in Copeland, 2004, 217-26; Mahon, 2004.

¹²⁸ Una delle opere letterarie preferite del piccolo Shannon era *Lo scarabeo d'oro* (*The Gold-Bug*) di Edgar Allan Poe, in cui si racconta di un uomo, William Legrand, che scopre un tesoro piratesco decifrando un codice trovato su una pergamena. Attraverso l'analisi della frequenza dei simboli e deduzioni logiche, Legrand risolve il mistero, illustrando un processo di crittoanalisi

idee mentre lavoravo sull'altro. Non direi che uno sia venuto prima dell'altro-erano così strettamente collegati che non si potevano separare [Shannon in Kanh, 1967, p.744]¹²⁹.

c. **Segretezza, ridondanza, entropia: crittoanalisi e teoria dell'informazione**

È possibile affermare che molte delle idee fondamentali del lavoro del 1948 trovarono ispirazione, e forse persino la loro configurazione peculiare, proprio nel contesto dei lavori crittografici e delle esperienze belliche di Shannon. Intere parti di *A Mathematical Theory of Communication* [Shannon, 1948] stando allo stesso autore ""Furono preso *verbatim* dal [rapporto declassificato del 1945] sulla crittografia, che non era stato pubblicato all'epoca"[Price, 1982]¹³⁰.

L'idea principale di Shannon è quella di studiare "il problema fondamentale delle comunicazioni" e cioè "quello di riprodurre esattamente o approssimativamente in un certo punto un messaggio scelto in un altro punto" [Shannon, 1948, p.33 trad. it.].

Shannon partì dallo stesso punto di riferimento che aveva guidato Harry Nyquist (1889-1976) e Ralph Hartley (1888-1970) nei loro studi sulla comunicazione, pubblicati negli anni Venti [Nyquist, 1924; Hartley, 1928]. Entrambi, ingegneri dei Bell Labs e consulenti del laboratorio statunitense durante il Progetto X, avevano proposto un approccio che "lasciasse da parte i fattori psicologici, per concentrarsi solo su quelli fisici" [Gleick, 2011, p.202 rad. it.].

Per far ciò nella "teoria dell'informazione" – "*information theory*" è una locuzione che fa capolino nella produzione di Shannon proprio nel suo lavoro sulla crittografia del '45 [Shannon, 1945, p.72] - bisogna eliminare tutto ciò che riguarda la dimensione *semantica* del messaggio. "Frequentemente i messaggi possiedono significato" prosegue il matematico americano, tuttavia "questi aspetti semantici della comunicazione sono irrilevanti per il problema tecnico."

¹²⁹ "During World War II," he has said, "Bell Labs were working on secrecy systems. I'd worked on communication systems and I was appointed to some of the committees studying cryptanalytic techniques. The work on both the mathematical theory of communications and the cryptology went forward concurrently from about 1941. I worked on both of them together and I had some of the ideas while working on the other. I wouldn't say one came before the other-they were so close together you couldn't separate them." Shannon vedeva il lavoro "pratico" di crittografia e crittoanalisi ["a very down-to-earth discipline" Soni, Goodman, 2017] come contraltare del suo, più teorico. In un rapporto di reciproca legittimazione e corroborazione. "I wrote [the information theory paper], which in a sense sort of justified some of the time I'd been putting into [cryptography], at least in my mind. . . . But there was this close connection. I mean they are very similar things. . . . Information, at one time trying to conceal it, and at the other time trying to transmit it." [Price, 1982].

¹³⁰ "Part of the 1948 paper "was taken verbatim from the [declassified 1945] cryptography report, which had not been published at that time." In Kline, 2015, p.33.

[Shannon, 1948, p.33 trad.it]. Quello che conta nella comunicazione è che “il messaggio effettivo viene scelto *tra un insieme* di possibili messaggi” [Shannon, 1948, p.33 trad.it].

Questa scarnificazione del contenuto semantico particolare ha, nel lavoro di Shannon, radici evidenti nel suo lavoro crittoanalitico: “Le uniche proprietà di una lingua di interesse nella crittografia sono le proprietà statistiche” [Shannon, 1945, p.10 trad. mia]¹³¹.

Il messaggio va *scelto*, e non *creato*, entro un spazio di probabilità e inviato da un punto all'altro riducendo, per quanto possibile (*supra*, lettera a Bush) il rumore”, rumore che era stato trattato in modo matematico anche dallo *Yellow Peril* wieneriano. Allo stesso l'obiettivo del crittografo è superare in astuzia l'analista crittografico nemico, “l'obiettivo dell'ingegnere delle comunicazioni è superare in astuzia il rumore” [Kline, 2015, p.34 trad. mia]¹³².

Affinché il messaggio possa essere scelto e la comunicazione possa concretizzarsi, “il sistema deve essere progettato per operare in relazione a qualunque scelta possibile, e non solo per quella che sarà effettivamente attuata, dal momento che questa non è nota in fase di progetto” [Shannon, 1948, pp. 33-34 trad. it]. Nel suo saggio sulla crittografia, Shannon aveva puntualizzato sin dall'introduzione che “un sistema di segretezza è definito in modo astratto come un insieme di trasformazioni di uno spazio (l'insieme dei messaggi possibili) in un secondo spazio (l'insieme dei crittogrammi possibili).” [Shannon, 1945, p.2]¹³³,

Ognuna di queste trasformazioni ha una “probabilità a priori associata ad essa”, la probabilità che sia scelta. L'intero spazio di probabilità delle trasformazioni, lo spazio dei messaggi, ha delle probabilità a priori [Shannon, 1945, p.2]¹³⁴.

¹³¹ “The only properties of a language of interest in cryptography are statistical properties”.

¹³² “If the goal of the cryptographer is to outwit the enemy cryptanalyst, the goal of the communications engineer is to outwit noise”.

¹³³ “A secrecy system is defined abstractly as a set of trans-formations of one space (the set of possible messages) into a second space (the set of possible cryptograms)”.

¹³⁴ “Each key and therefore each transformation is assumed to have an a priori probability associated with it-the probability of choosing that key. The set of messages or message space is also assumed to have a priori probabilities for the various messages, i.e., to be a probability or measure space.” Il decrittatore, una volta intercettato il crittogramma, calcola le probabilità a posteriori dei possibili messaggi e chiavi, basandosi sulle probabilità a priori associate sia alle chiavi sia ai messaggi. Queste probabilità a priori rappresentano la conoscenza iniziale dell'analista crittografico nemico riguardo alle possibili scelte, come le frequenze delle lettere in una lingua normale (come l'inglese o l'italiano).

Nel caso dell'*inglese* con 26 lettere, ci sono 26! possibili trasformazioni, che equivalgono a circa 4×10^{26} combinazioni. Questo grande numero rende il sistema sicuro fintanto che il nemico dispone di poche informazioni. Se il nemico intercetta N lettere di crittogramma in questo sistema, le sue probabilità cambiano. Se il crittografo nemico intercetta:

- $N=50$: Le frequenze tipiche delle lettere in inglese, come “e” o “t”, permettono all'analista di identificare un messaggio con probabilità a posteriori vicina a 1, rendendo il crittogramma decifrabile con precisione.

- $N=15$: Rimangono molteplici messaggi plausibili con probabilità simili, quindi il crittogramma non fornisce una soluzione unica.

Un sistema crittografico è composto da un insieme limitato (anche se potenzialmente molto ampio) di messaggi disponibili, un insieme definito di crittogrammi ottenibili e, come collegamento tra i due, un gruppo finito di chiavi. “nel mezzo, per trasformare gli uni negli altri, un insieme finito di chiavi, a ciascuna delle quali è associata una probabilità” [Gleick, 2011].

Shannon [Fig.6] ci offre nel saggio sulla crittografia una rappresentazione grafica del sistema di comunicazione.

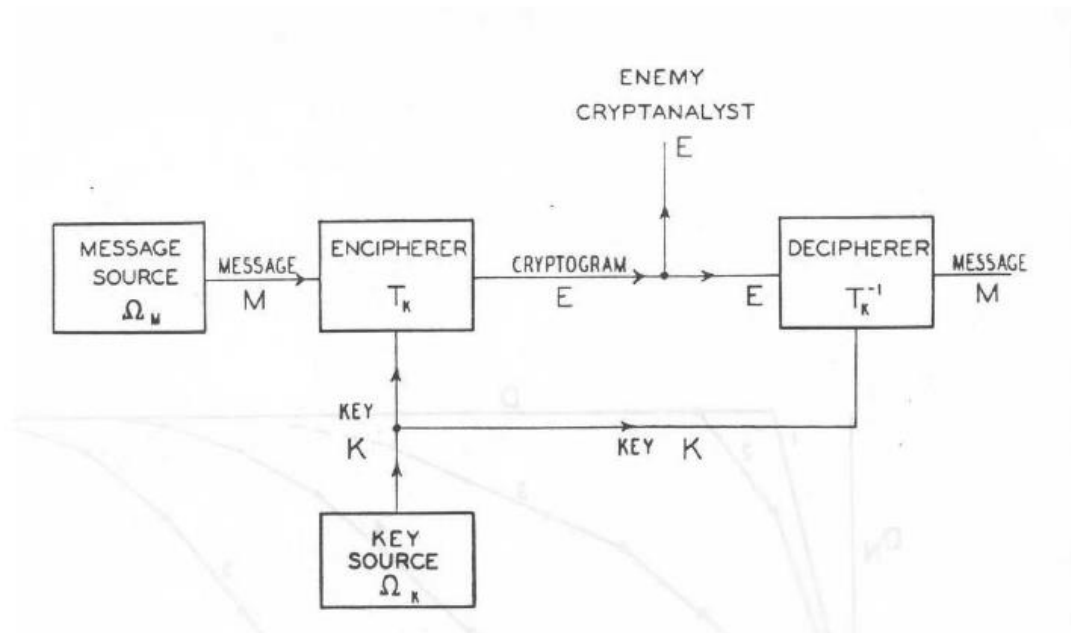


Fig.6 in *A Mathematical Theory of Cryptography*, Shannon, 1945, p.116.

SI può provare a fare lo stesso calcolo con le lettere comunemente usate nell'italiano corrente. Nel caso dell'italiano, che usa 21 lettere, ci sono $21!$ combinazioni, pari a circa 5×10^{19} . Anche qui, il numero di lettere intercettate è cruciale:

-N=50: Le lettere italiane più comuni, come "e", "a" e "i", aiutano a identificare il messaggio corretto, poiché le loro frequenze riducono le ambiguità.

-N=15: L'analista nemico avrebbe ancora molte possibilità, senza che una soluzione emerga con alta probabilità.

La "sicurezza perfetta", infine, si ottiene quando le probabilità a posteriori dei messaggi, dopo l'intercettazione, coincidono con quelle a priori. In questo caso, il crittogramma non rivela alcuna informazione utile al nemico. Tuttavia, per ottenere questa sicurezza, è necessario che il numero di chiavi sia almeno pari al numero di messaggi possibili, un requisito che rende il sistema impraticabile nella maggior parte dei contesti. [Shannon, 1945, pp-2-5]. Come spiega James Gleick a proposito della fondazione scientifica della crittografia da parte di Shannon, il matematico americano “ha dimostrato che sono possibili cifrari perfetti, “perfetti” nel senso che anche l'intercettazione di un messaggio di lunghezza infinita non sarebbe d'aiuto a chi tentasse di violarlo (“Il nemico, dopo aver intercettato una quantità qualsiasi di materiale, non si trova in una posizione migliore di prima”). Ma, come dava, toglieva anche, perché dimostrava che i requisiti erano così forti da risultare praticamente inutili. In un cifrario perfetto, tutte le chiavi debbono essere egualmente probabili, in effetti devono essere un flusso casuale di caratteri; ogni chiave può essere usata solo una volta; e, peggio di tutto, ogni chiave deve essere lunga quanto l'intero messaggio” [Gleick, 2011, p.202 trad. it.]

Questa rappresentazione è la versione crittoanalitica [Fig.7] dello schema generale della comunicazione, offerto poi in *A Mathematical Theory of Communication*:

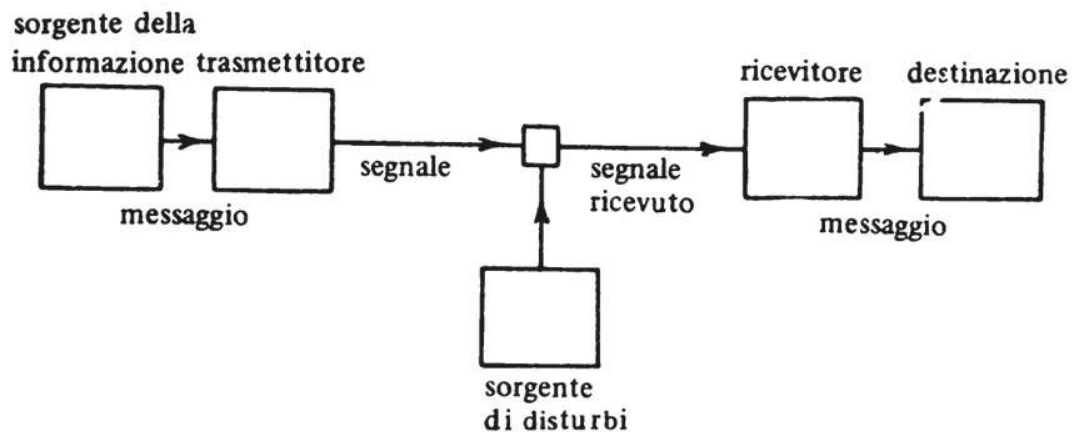


Fig.7 in *A Mathematical Theory of Communication*, Shannon, 1948, p.36. trad. it.

Nello schema, gli elementi individuati da Shannon sono:

- 1) *Sorgente di informazione*: Genera il messaggio da trasmettere, che può essere una sequenza di lettere, una funzione temporale (es. audio) o una combinazione complessa come nella televisione a colori (es. funzioni di spazio, tempo e intensità luminosa).
- 2) *Trasmettitore*: Trasforma il messaggio in un segnale adatto alla trasmissione. Ad esempio, converte la pressione sonora in corrente elettrica (telefonia) o codifica segnali in punti e linee (telegrafia).
- 3) *Canale*: Il mezzo fisico che trasporta il segnale dal trasmettitore al ricevitore, come fili, cavi coassiali, onde radio o fasci di luce. Durante il percorso, il segnale può essere influenzato da disturbi.
- 4) *Ricevitore*: Esegue l'operazione inversa del trasmettitore, decodificando il segnale ricevuto per ricostruire il messaggio originale.

- 5) *Destinazione*: È l'utente finale (persona o dispositivo) a cui il messaggio è diretto.
[Shannon, 1948, p.37 trad. it.]¹³⁵

Si paragonino i due schemi: l'analista di codici nemico e il destinatario di una comunicazione generica condividono lo stesso obiettivo, ossia il messaggio. L'eredità del lavoro di crittoanalisi permise a Shannon di individuare la chiave per matematizzare con rigore la quantità di informazione e di compiere così uno scarto rispetto alla sola intelligence, che fino ad allora aveva assorbito le sue riflessioni – come testimonia la lettera a Bush del 1939 – inquadrando il messaggio in termini statistici all'interno di un insieme più ampio di alternative possibili (indicando la probabilità dell'evento in questo spazio con p_1).

Così come l'analisi probabilistica, la base dell'approccio matematico di Wiener intorno ai sistemi a retroazione, aveva permesso di superare la dicotomia tra organico e artificiale, astruendo da ogni differenza strutturale tra i vari tipi di sistemi, allo stesso modo l'analisi probabilistica di Shannon, applicata alla teoria della comunicazione, consentì di oltrepassare il confine tra artificiale e organico, tra umano e macchina, giungendo a una nozione onnicomprensiva della comunicazione. Entrambi gli approcci furono profondamente plasmati dalle condizioni belliche.

Ne è una testimonianza diretta quanto Warren Weaver, il matematico dell'NDRC supervisore di Wiener e di Shannon durante la guerra, scrive nella sua nota introduttiva ad *A Mathematical Theory of Communication*:

Il termine comunicazione sarà da noi usato in un senso molto ampio per comprendervi tutti i procedimenti attraverso i quali un pensiero può influenzarne un altro. Questi, naturalmente, comprendono non solo il linguaggio scritto e parlato, ma anche la musica, le arti figurative, il teatro, la danza e, di fatto, qualunque comportamento umano. In qualche caso può risultare preferibile usare una definizione ancora più vasta di comunicazione, vale a dire una definizione che includa i procedimenti attraverso i quali un meccanismo (come un impianto automatico per il rilevamento di un aeroplano e per il calcolo della sua probabile posizione futura) entra attivamente in rapporto con un altro meccanismo (un missile diretto alla caccia dell'aeroplano) [Weaver, 1949, p. 1 trad. it].

¹³⁵ Shannon cita ripetutamente descrivendo gli elementi del suo schema di comunicazione il sistema PCM del Vocoder, tecnologia chiave del Progetto X, che, come visto sopra, presupponeva la meccanizzazione/quantizzazione e la campionatura del parlato. Shannon, 1948, pp.36-37. L'importanza del lavoro sulla tecnologia alla base del Vocoder sulla teoria dell'informazione è analizzata in Strane, 2012. Sebbene, come nota Kline, questa analisi dettagliata -che contribuisce a dare la misura della pervasività dell'eredità nella nostra epoca delle idee e delle tecnologie ad esse ispirate che hanno preso forma nel periodo bellico – manca di cogliere appieno il nesso tra PCM e crittografia di guerra [Kline, 2015, p.260 n.10].

Ciò che distingue il lavoro di Shannon dai precedenti tentativi di fornire una fondazione matematica rigorosa alla nozione di quantità di informazione è l'aver tenuto conto della dimensione *probabilistica* con cui i simboli vengono selezionati

Ralph Hartley aveva preso in considerazione la relazione combinatoria tra la quantità di simboli trasmessi e la dimensione dell'insieme da cui sono presi. Quanto maggiore è il numero di simboli possibili, tanto maggiore è la quantità di informazione che porta ciascuna selezione, ma la relazione è logaritmica: per raddoppiare la quantità di informazione, è necessario elevare al quadrato le dimensioni dell'alfabeto. Hartley formalizzò l'informazione come:

$$H = n \log s \text{ [Hartley, 1928]}$$

Dove H è la quantità di informazione, n il numero di simboli trasmessi e s la dimensione dell'insieme, ad esempio un alfabeto.

Tuttavia, Hartley non considerò le probabilità con cui i simboli venivano scelti, trattando ogni simbolo come equiprobabile. Questa semplificazione, benché utile per sistemi come il codice Morse o i telegrafi [Gleick, 2011, pp.185-187 trad. it.], ignorava la struttura statistica delle sorgenti reali di informazione. Shannon invece supera questa limitazione considerando nell'essenza di ogni struttura di comunicazione la probabilità con cui i simboli sono scelti, e definisce la quantità di informazione con la formula [Shannon, 1948, p.54-55 trad. it.], che compare appare all'inizio di *A Mathematical Theory of Cryptography* [Shannon, 1945, p.7]:

$$H = - \sum_{i=1}^n p_i \log p_i$$

Nella formula di Shannon, H , la quantità di informazione rappresenta “quanta informazione è generata quando la scelta è fatta” [Shannon, 1945, p.9 trad. mia]¹³⁶.

In altre parole, essa misura la libertà di scelta nella selezione di un messaggio, e cioè l'incertezza associata a tale scelta. “Nessuna incertezza - nessuna libertà di scelta - nessuna informazione” [Weaver, 1948, p.16 trad. it.]. Questa incertezza può essere calcolata considerando il numero dei messaggi possibili, evidenziando così la relazione diretta tra la varietà delle alternative e la quantità di informazione generata. Quando una probabilità è

¹³⁶ “H measures in a certain sense how much “information is generated” when the choice is made”.

uguale a 1 (cioè rappresenta la certezza), tutte le altre sono 0. Non esiste libertà di scelta: non c'è alcuna incertezza da risolvere e, di conseguenza, nessuna informazione da trasmettere.

A questa quantificazione dell'incertezza Shannon dà il nome, evocativo, di *entropia*¹³⁷. Questa aveva bisogno di un'unità di misura. Qui fa la sua entrata in scena il *bit*¹³⁸. "L'unità di informazione è detta *bit*, termine proposto da John W. Tukey [collega di Shannon ai Bell Labs] in luogo dell'espressione completa *binary digit* (cifra binaria)". [Shannon, 1948, p.9 trad. it.].

Nella teoria matematizzata di Shannon, gli elementi della comunicazione - Sorgente di informazione, Trasmettitore, Canale, Ricevitore e Destinazione - nel caso della comunicazione umana "questi elementi sono il cervello del parlante, le corde vocali del parlante, l'aria, l'orecchio dell'ascoltatore e il cervello dell'ascoltatore" [Gleick, 2011, p. 206]. Shannon, eliminando ogni riferimento alla semantica e quantificando la comunicazione in relazione all'entropia, aveva definito lo schema di qualunque entità capace di processare informazione, sia essa artificiale o umana, anche nella loro relazione reciproca o integrata, radicalizzando il passo di equiparazione funzionale a livello logico tra cervelli e componenti dei calcolatori come i relè o i circuiti flip-flop [Shannon, 1948, p. 35 trad. it.].

Da questo punto di vista, ed eliminato ogni riferimento alla natura semantica dei messaggi, cosa rimane? Cosa rimane di sequenze discrete di simboli emesse da una sorgente, come una lingua naturale?

I linguaggi non sono altro che processi stocastici. Questo tipo di processo non è né deterministico, né totalmente causale. È invece quel tipo di processo, statistico, "che genera una sequenza di simboli secondo un determinato sistema di probabilità" considerando come

¹³⁷ Ai Bell Labs si raccontava che John von Neumann avesse suggerito a Claude Shannon di usare il termine "entropia" perché avrebbe vinto ogni discussione, dato che nessuno lo avrebbe compreso. Tuttavia, questa storia è probabilmente falsa, poiché Shannon utilizzò il termine solo una volta nel suo rapporto di crittografia del 1945 (Shannon, 1945, p. 10). In questa accezione l'entropia e l'informazione hanno, come dimostra la fermezza della posizione di Shannon nell'ignorare i significati particolari dei messaggi, una natura soltanto sintattica. Nel corso di questo lavoro, trattando gli aspetti fondazionali di un'etica dell'informazione, si considereranno l'informazione e l'entropia da un punto di vista anche semantico (più vicina alla considerazione wieneriana dell'informazione) [Kline, 2015, p.13; Floridi, 2009, p.37].

¹³⁸ Un bit, come minima quantità di informazione, in accordo con la teoria matematica di Shannon, rappresenta l'incertezza associata ad una alternativa binaria (come può essere 01 o on/off o VF). Si può fare l'esempio del lancio di una moneta. Questo esperimento implica una scelta tra due eventi equiprobabili ($p_1 = p_2 = \frac{1}{2}$). Applicando la formula di Shannon, $H = -\sum_{i=1}^n p_i \log p_i$ il calcolo è:

$$H = -\left(\frac{1}{2} \log_2 \frac{1}{2} + \frac{1}{2} \log_2 \frac{1}{2}\right) = -\left(\frac{1}{2}(-1) + \frac{1}{2}(-1)\right) = 1 \text{ bit}$$

Quindi, il lancio di una moneta produce 1 bit di informazione, rappresentando la completa incertezza tra due alternative ugualmente probabili. Altri esempi accessibili, senza l'esplicitazione esemplificativa del calcolo qui mostrata, sono in Gleick, 2011.

simboli le lettere delle parole del linguaggio, gli spazi e le punteggiature [Shannon, 1945, p.11 trad. mia]¹³⁹.

Si tenga in mente la struttura di ogni comunicazione proposta da Shannon, in essa “ogni processo stocastico sufficientemente complesso fornirà una soddisfacente rappresentazione di una sorgente discreta. [Shannon, 1948, p.48 trad. it.]. Questa rappresentazione (la sorgente) va considerata come “un generatore del messaggio simbolo dopo simbolo. Essa sceglierà i simboli successivi in conformità a certe probabilità dipendenti, in generale, dalle scelte precedenti come anche dai particolari simboli in questione” [Shannon, 1948, p.42 trad. it.]. La quantità di informazione, l'entropia, sarà in relazione con probabilità di occorrenza di questi simboli discreti.

Il rapporto tra l'entropia di una sorgente e il valore massimo che essa potrebbe avere, limitatamente agli stessi simboli, sarà detto entropia relativa della sorgente. Questa, come si vedrà in seguito, costituisce la massima compressione possibile quando procediamo alla codificazione con un medesimo alfabeto. Uno [la probabilità massima] meno l'entropia relativa costituisce la ridondanza. [Cfr. Shannon, 1948, p.61 trad. it.].

Ogni lingua, come ben sapevano gli operatori del telegrafo, possiede una sua ridondanza. “Per fare un esempio semplice, in inglese come in italiano, ogni volta che compare una q, la u che la segue è ridondante. [...] Dopo una q, ci si aspetta una u. Non c'è sorpresa, non convoglia alcuna informazione” [Gleick, 2011, p.200 trad. it.]. Shannon mostrò che la lingua inglese ha una ridondanza del 50% [Shannon, 1948, p.61 trad. it.].

Come commenta Weaver, “circa metà delle lettere o delle parole che scegliamo nello scrivere e nel parlare dipendono dalla nostra libera scelta, e circa metà (sebbene normalmente

¹³⁹ “We consider a language, therefore, to be a stochastic (i.e., a statistical) process which generates a sequence of symbols according to some system of probabilities. The symbols will be the letters of the language, together with punctuation, spaces etc., if these occur.” Di converso ogni processo di generazione stocastica di simboli può essere considerato un linguaggio. Tra questi possono essere ricompresi i casi di: 1) Linguaggi scritti naturali; 2) sorgenti di informazione continue che sono state rese discrete attraverso un processo di quantizzazione (es. il segnale PCM che quantizza il parlato umano o un segnale televisivo); 3) linguaggi “artificiali” basati su processi stocastici per generare sequenze di simboli. Esistono diversi tipi di linguaggi artificiali, ognuno basato su processi stocastici e con livelli di complessità differenti. Alcuni generano simboli in modo del tutto indipendente, scegliendoli casualmente con probabilità fisse. Altri introducono una dipendenza tra i simboli, dove la scelta di ciascun simbolo dipende da uno o più simboli precedenti, come nel caso dei digrammi o dei trigrammi. Un ulteriore livello di complessità si raggiunge con linguaggi composti da sequenze di parole, in cui ogni parola ha una probabilità definita e possono esserci transizioni tra parole successive. Questi modelli aiutano a rappresentare e analizzare diverse caratteristiche dei linguaggi naturali, semplificandone alcune dinamiche [Shannon, 1945, pp.11-14]. L'analisi di Shannon sulle probabilità stocastiche *markoviane* dei linguaggi è tutt'ora di strettissima attualità vista la pervasività contemporanea di tecnologie algoritmiche come gli LLMs (Large Language Models) che possono essere considerati dei “pappagalli stocastici” [Bender et al., 2021]. Se l'intelligenza artificiale odierna ha natura probabilistica, a questa visione, come vedremo più avanti si opposero pensatori chiave del paradigma dell'IA simbolica come Chomsky.

non ne siamo consapevoli) sono in realtà controllate dalla struttura statistica del linguaggio” [Weaver, 1949, p.14 trad. it.]. La parte ridondante, pur se mancante, consentirebbe comunque di completare, come nelle parole crociate il messaggio. Ed essa è ad entropia =0, in quanto non informa e non dipende dalla scelta del messaggio¹⁴⁰

Shannon dimostrò la correlazione tra entropia, informazione e ridondanza mediante un esperimento di progressiva approssimazione: partendo da un linguaggio artificiale, giunse gradualmente a una lingua naturale. Servendosi dell’alfabeto inglese, evidenziò i diversi gradi di dipendenza probabilistica di un simbolo rispetto al numero di simboli precedenti, un approccio che compare sia nel suo lavoro del 1945 sulla crittografia, sia in *A Mathematical Theory of Communication*, sia nell’intervento presentato alle Macy Conferences del marzo 1950, in cui le sue idee sul linguaggio si diffusero all’interno della comunità cibernetica

1. Approssimazione di grado zero (simboli indipendenti ed ugualmente probabili).
XFOML RXKHRJFFJUJ ZLPWCFWKCYJ FFJEYVKCQSG HYD QPAAMKBZAACIBZLHJQD.
2. Approssimazione di primo grado (simboli indipendenti ma con frequenze proprie della lingua inglese).
OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI ALHENHTTPA OOBTTVA NAH BRL.
3. Approssimazione di secondo grado (struttura dei digrammi come in inglese).
ON LE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY ACHIN D ILONASIVE TUOOOWE AT TEASONARE FUSO TIZIN ANDY TOBE SEACE CTISBE.
4. Approssimazione di terzo grado (strutture dei trigrammi come in inglese).
IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID PONDENOME OF DEMONSTURES OF THE REPTAGIN IS REGOACTIONA OF CRE.
5. Approssimazione di primo grado, con riguardo alle parole.
REPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT NATURAL HERE HE THE A IN A CAME THE TO OF TO EXPERT GRAY COME TO FURNISHES THE LINE MESSAGE HAD BE THESE.
6. Approssimazione di secondo grado, con riguardo alle parole.
THE HEAD AND IN FRONT AL ATTACK ON AN ENGLISH WRITER THAT THE CHARACTER OF THIS POINT IS THEREFORE ANOTHER METHOD FOR THE LETTERS THAT THE TIME OF WHO

¹⁴⁰ Qui si può notare nuovamente la strettissima connessione delle teorie sulla comunicazione di Shannon e la crittoanalisi. “Nella crittoanalisi, quella stessa ridondanza è il tallone di Achille” [Gleick, 2011, p.200] in quanto la ridondanza permette di ricostruire il messaggio a partire dalle occorrenze dei simboli dipendenti dalla struttura statistica della lingua. Come nota lo storico della cibernetica, R. Kline, dalla prospettiva dell’informazione di Shannon “L’equivocazione era quasi identica all’informazione, la differenza risiedeva soltanto nel punto di vista” [“Equivocation was almost identical with information, the difference being one of point of view, Kline, 2015, p.35 trad. mia.].

EVER TOLD THE PROBLEM FOR AN UNEXPECTED. [Shannon, 1945, pp.14-15; 1948, pp.47-48 trad. it; Shannon 1950, pp.249].

Come mostra il punto 6, “è chiaro dunque che un processo stocastico sufficientemente complesso fornirà una soddisfacente rappresentazione di una sorgente discreta” [Shannon, 1948, p.48 trad. it.]. L’esperimento, inoltre, permette di calcolare la ridondanza¹⁴¹ e, per converso, l’entropia di una lingua come l’inglese, che all’approssimazione 5 presenta una ridondanza del 54%, mostrando quanto la sua struttura statistica lo renda prevedibile. [Cfr. Shannon 1950, p.250].

Durante l’esposizione Shannon, appena dopo aver affermato che la “ridondanza della riga B fornisce una stima della ridondanza del testo originale, cioè dell’inglese.” [Shannon, 1950, p.252 trad. mia]¹⁴², fu subito interrotto da Wiener che fece notare le somiglianze tra la metodologia di Shannon e quella il suo lavoro sulla predizione¹⁴³, sviluppato nello *Yellow Peril*, facendo notare come i due metodi avessero però aspetti paralleli¹⁴⁴.

¹⁴¹ Il lavoro di Shannon sulla ridondanza e sull’entropia ha permesso al matematico statunitense di giungere a una serie di risultati fondamentali riguardo l’ingegneria delle comunicazioni. Primo tra tutti il “teorema fondamentale per un canale non disturbato”, il teorema numero 9 di *A Mathematical Theory of Communication* [Shannon, 1948, p.64 trad. it] che afferma in un canale non disturbato con capacità C (in bit al secondo) e una sorgente con entropia H (in bit al secondo), è possibile trasmettere simboli con velocità media fino a C/H usando opportune codifiche, $velocità\ media = \frac{C}{H}$ m a mai superarla.. La formula principale è: $velocità\ media = \frac{C}{H}$. In presenza di disturbi, la ridondanza diventa velocità media vantaggiosa poiché aiuta a correggere gli errori. “Quando vi sono disturbi nel canale, [...] vi è qualche vantaggio effettivo nel non usare un processo di codifica che elimini del tutto la ridondanza. Infatti, la restante ridondanza serve a combattere i disturbi” [Weaver, 1949, p.23 trad. it.]. È possibile che il mittente debba usare più bit per correggere gli errori, rallentando la trasmissione, ma garantendo che il messaggio arrivi comunque a destinazione.

¹⁴² “From the data in the second line B, it is possible to set upper and lower bounds for the entropy of English. There is a theorem on stochastic processes that the redundancy of a translation of a language is identical with that of the original, if it is a reversible translating process going from the first to the second. Consequently, an estimation of the redundancy of the line B gives an estimate of the redundancy of the original text, that is, of English. Line B is much easier to estimate than line A, since the probabilities are more concentrated. The symbol 1 has a very high individual probability, and the symbols from 6 to 27 have very small probabilities”

¹⁴³ Bigelow ricorda il rapporto tra Wiener e Shannon: “Vedevo Wiener dare in continuazione a Shannon consigli, aiuto e idee. Shannon arrivava e gli parlava di idee di comunicazione e di come esprimerle, e Wiener faceva associazioni libere e voli pindarici. Credo che Wiener sia stato il sostegno delle opinioni di Shannon e di molte delle sue idee sulla teoria dell’informazione”. Wiener dopo un po’ tuttavia iniziò a risentirsi delle visite di Shannon. Wiener “era molto gentile” disse, “ma quando Claude Shannon arrivava al MIT, - come ricorda Fagi Levinson, moglie del matematico del MIT e allievo di Wiener, Norman Levinson - commentava: “È venuto a saccheggiarmi il cervello.” Non lo voleva vedere, lo evitava. Temeva che altri gli rubassero le idee e si prendessero il merito del suo lavoro” in Conway, Siegelman, 2005, pp.185-186 trad. it. e Kline, 2015, p.31.

¹⁴⁴ “This is very interesting to me. In my prediction theory, I start from a series with a correlation and I build a series which is equivalent; each is the past linear function of the past of the other. In addition, however, in my new series the choices are completely independent, whereas in my earlier series the series are partially dependent. In fact, the way I built the linear prediction theory was by the reduction of my dependent choices to independent choices. My method has some parallelisms to this. Excuse me for interrupting”.

Wiener stesso, nella sua autobiografia, definì la formula sulla quantità di informazione, la “definizione Shannon-Wiener della quantità di informazione (poiché essa appartiene egualmente a entrambi)” [Wiener, 1953b, p.263 trad. mia]¹⁴⁵, data l’importanza del concetto – e di quello correlato di entropia - nel suo itinerario intellettuale [Wiener, 1945; 1948 Capp.II e III pp.73-132 trad. it; 1950, pp.34-73 trad.it.]¹⁴⁶.

Ma erano esattamente le stesse? Malgrado delle dichiarazioni di sostanziale accordo [Kline, 2015, p.16] le due versioni della definizione divergevano sul segno da assegnare all’entropia, alle caratteristiche dell’informazione e al residuo semantico. Per Shannon, come abbiamo visto, l’informazione è entropia *positiva*, in quanto è misura dell’incertezza, del disordine e quindi della informatività di una comunicazione. Non c’è informazione, né si genera, quando la selezione del messaggio è nota.

Wiener invece era di parere opposto¹⁴⁷, in *Cybernetics*, uscito nello stesso anno di *A Mathematica Theory of Communication*, scrive: “Come la quantità di informazione in un sistema è una misura del suo grado di organizzazione, così l’entropia di un sistema è una misura del suo grado di disorganizzazione ; e l’una è semplicemente l’opposto dell’altra” [Wiener, 1948, p.34 trad. it], concetto che ripeterà due anni dopo in *The Human use of Human Beings*¹⁴⁸.

Per Wiener l’informazione è entropia negativa. Se Shannon aveva tratto la sua nozione di informazione dalla crittografia, Wiener vi era giunto attraverso il lavoro sul predittore, la conseguente caratterizzazione di ogni sistema come struttura finalizzata a retroazione negativa e la matematizzazione dei fenomeni probabilistici. Shannon rimase su un piano puramente matematico, al contrario di Wiener che aveva “interpretato l’entropia in maniera fisica” [Kline, 2015, p.24]¹⁴⁹ legata al senso al senso termodinamico del termine. “L’informazione misura

¹⁴⁵ “In introducing the Shannon-Wiener definition of quantity of information (for it belongs to the two of us equally), we made a radical departure from the existing state of the subject”.

¹⁴⁶ La questione intorno a chi, tra Shannon e Wiener, spettasse “primato” della teoria della quantità di informazione, che coinvolge una fuga in Messico di Walter Pitts, è raccontato in Kline, 2015, pp.9-18.

¹⁴⁷ Per la storia concettuale dello sviluppo del concetto di “quantità di informazione” nel pensiero di Wiener, si veda Kline, 2015, pp.22-26.

¹⁴⁸ “Il concetto di entropia è associato a quello di *modello* ed esprime la quantità di disordine in una data classe di modelli. Inoltre esso è strettamente connesso con il concetto di *informazione* e della sua misura, che è essenzialmente una misura di ordine. La quantità di informazione è una misura di quel grado di ordine che è particolarmente associato a quei modelli distribuiti nel tempo che sono i messaggi”. Wiener, 1950, p.35 trad. it.]

¹⁴⁹ “Unlike Shannon, however, Wiener interpreted entropy in a physical manner”. Inoltre bisogna sottolineare come, malgrado il rispetto formale dimostrato reciprocamente tra Shannon e Wiener nel riconoscere i meriti dell’altro nel campo della teoria dell’informazione, [Wiener, 1948, p.34 trad.it e 2017; Shannon, 1948, p.126 trad.it; Kline, 2015, p.36] i due erano divisi da una impostazione “filosofica” di fondo. Afferma Wiener nella sua autobiografia “Shannon ama il discreto e rifugge dal continuo. Considerava i messaggi discreti come una sorta di sequenza di sì e no distribuiti nel tempo, e vedeva ogni singola decisione tra sì e no come l’elemento dell’informazione. Nella teoria continua del filtraggio, ero stato condotto a una definizione molto simile dell’unità di informazione, partendo inizialmente da un punto di vista considerevolmente diverso” [“Shannon loves the

l'ordine e l'entropia misura il disordine" [Wiener, 1948b, p.203]¹⁵⁰: L'informazione di un messaggio poteva essere solo degradata.

L'idea di informazione come entropia negativa, che attraversa l'intero pensiero di Wiener, spinse il matematico del MIT a richiamarsi – a riprova della centralità del feedback, guidato dall'informazione, in ogni sistema, sia esso artificiale o naturale – nelle pagine introduttive di *Cybernetics* [Wiener, 1948, p. 34 trad. it.], alle concezioni biologiche del fisico Erwin Schrödinger (1897-1961) e alla sua opera *What Is Life?* [1944]. Per lo scienziato austriaco, un organismo poteva mantenersi in vita soltanto attingendo dall'ambiente entropia negativa, per sottrarsi allo stato di entropia massima, ossia alla morte. Ogni sistema, compresi quelli biologici, utilizza dunque l'informazione per resistere alla degradazione entropica. Nel pensiero wieneriano, ciò si applica anche alla sfera sociale, come vedremo più avanti, quando esploreremo la dimensione etica e i fondamenti delle tecnologie digitali e dell'informazione. Queste considerazioni ci consentono di intravedere una prima risposta alla domanda "controculturale" posta all'inizio di questo lavoro dalla poesia di Brautigan

Wiener sviluppò queste idee di carattere più speculativo – collaborando anche con il filosofo di origine italiana Giorgio De Santillana (1902-1974) – nel capitolo di apertura di *Cybernetics, Newtonian and Bergsonian Time* [Wiener, 1948, pp. 56-72 trad. it.]. L'affermarsi della meccanica statistica aveva infatti superato la concezione "newtoniana" della fisica classica, fondata sull'idea di un tempo reversibile [Wiener, 1948, p. 64 trad. it.]. Gli stessi fenomeni fisici, alla luce dei concetti termodinamici, devono essere studiati entro un tempo "bergsoniano", quel medesimo tempo che il filosofo francese riteneva pertinente soltanto al mondo della vita [Wiener, 1948, p. 72 trad. it.].

La cibernetica studiando macchine in cui vi è una relazione tra input e output che prevede una unidirezionalità del tempo, studia macchine che appartengono allo stesso tempo irreversibile della biologia. La comunicazione, tra uscita ed entrata, è un fenomeno connesso

discrete and eschews the continuum. He considered discrete messages as something like a sequence of yeses and noes distributed in time, and he regarded single decisions between yes and no as the element of information. In the continuous theory of filtering, I had been led to a very similar definition of the unit of information, from what was at the beginning a considerably different point of view", Wiener, 1953b, p.263 trad. mia]

¹⁵⁰ "In fact, it is not surprising that entropy and information are negatives of one another. Information measures order and entropy measures disorder". Queste considerazioni, apparse in *Time, Communications and Nervous System*, paper del 1948 [Wiener, 1948b] scritto per un intervento che Wiener doveva tenere alle Macy Conferences, si ritrovano in *Cybernetics* (dello stesso anno), dove il legame con la termodinamica e la fisica statistica, sempre al centro degli studi di Wiener, viene esplicitato con maggior chiarezza. Questo punto [l'entropia come misura del disordine e l'informazione come misura dell'ordine] di vista "ci condusse, ad alcune considerazioni sulla seconda legge della termodinamica, e a uno studio sulla possibilità dei cosiddetti demoni di Maxwell" [Wiener, 1948, p.34 trad.it.]

con l'entropia, e quindi, come abbiamo visto in precedenza, "la teoria di questa macchina appartiene così alla meccanica statistica *gibbsiana* piuttosto che alla meccanica classica newtoniana" [Wiener, 1948, p. 72 trad. it.].

Gli automi, le macchine studiate dalla cibernetica, la cui struttura fondamentale è quella di essere sistemi di feedback negativo, sono nello stesso tempo dei sistemi biologici e di tutti gli altri tipi, il tempo unidirezionale, statistico, in cui l'entropia, a cui si oppone l'informazione, è destinata a crescere:

L'automa moderno è quindi nello stesso tipo di tempo bergsoniano dell'organismo vivente, e di conseguenza non vi è alcuna ragione nelle considerazioni bergsoniane per cui le modalità di funzionamento essenziali dell'organismo vivente non debbano essere uguali a quelle dell'automa di questo tipo [Wiener, 1948, p.72 trad. it.]¹⁵¹.

Nel tempo bergsoniano, si apre la possibilità di una relazione informazionale tra sistemi, inclusa quella tra uomo e macchina nell'ambito della *Human-Computer Interaction* (HCI). Tale relazione mira a gestire, attraverso le interfacce, l'entropia generata dall'aumento della complessità informativa. Non a caso, T. Bardini sottolinea il carattere fondamentale del passaggio concettuale nelle concezioni temporali introdotto da Wiener nel suo studio sull'opera di un pioniere dell'HCI, Douglas Engelbart (1925-2013), la cui eredità continua a influenzare in modo decisivo la relazione uomo-macchina e l'azione umana mediata dalla tecnologia. Automi ed essere umani potevano interagire perché erano nello stesso tempo [Bardini, 2000, p.48-54].

d. "Fa' come Wiener: aggiusta il tiro": cervelli informazionali e liberazione individuale

Vi è un altro aspetto cruciale che emerge dall'analisi del lavoro sulla quantità di informazione, e sui suoi prodromi, che è necessario considerare. Nel 1943 Warren McCulloch (1898 – 1969) e Walter Pitts (1923 – 1969) - il primo, poliedrico neuropsicologo ed instancabile animatore del movimento cibernetico e presidente delle *Macy Conferences* ed il secondo precocissimo *enfant prodige* già allievo di Carnap e che collaborerà anche con Wiener -

¹⁵¹ Per approfondire sulle idee intorno alla *Anisotropia* del tempo di Wiener e la relazione del concetto di tempo con l'entropia (analizzata a anche dal punto di vista matematico) e gli altri concetti fondamentali della cibernetica, Masani, 1990, pp.151-157.

pubblicarono una pietra miliare della cibernetica e della scienza cognitiva, *A Logical Calculus of the Ideas Immanent in Nervous Activity* [McCulloch, Pitts, 1943]. Questo lavoro – che incontreremo durante tutto il corso del nostro lavoro - introdusse un modello astratto di neurone [McCulloch, 1974] concepito come un'entità formale che rappresentava proposizioni logiche. “A causa del carattere "tutto-o-niente" dell'attività nervosa, gli eventi neuronali e le relazioni tra di essi possono essere trattati attraverso la logica proposizionale” [McCulloch, Pitts, 1943, p.99 trad. mia]¹⁵².

Nel modello di McCulloch e Pitts le relazioni fisiologiche tra le attività nervose corrispondevano alle relazioni della logica proposizionale (sviluppate in particolar modo da Carnap). In questo lavoro i due cibernetici statunitensi mostrarono, pur non dimostrandolo, come queste reti potevano realizzare qualsiasi funzione logica computabile organizzando i neuroni in circuiti logici.

Il calcolo logico era implementato attraverso configurazioni specifiche: la relazione di *precedenza temporale* (*precession*) era determinata da configurazioni in cui un neurone si attivava solo se un altro neurone aveva trasmesso un segnale nel momento immediatamente precedente; la *disgiunzione* (*disjunction*) era realizzata con neuroni che rispondevano a segnali provenienti da una qualsiasi delle loro connessioni; la *congiunzione logica* (*conjunction*) era rappresentata da neuroni che si attivavano solo quando ricevevano segnali eccitatori simultanei sufficienti; la *negazione congiunta* (*conjoined negation*), infine, si otteneva attraverso sinapsi inibitorie che bloccavano l'attivazione del neurone.

Questa formalizzazione [fig.8] ha consentito di modellizzare il comportamento delle reti come una serie di operazioni logiche eseguibili sulla base delle connessioni sinaptiche e delle soglie di attivazione dei neuroni [McCulloch, Pitts, 1943].

¹⁵² Per una analisi puntuale, filosoficamente rigorosa e scientificamente accurata, del seminale lavoro di McCulloch e Pitts si suggerisce Piccinini, 2004. Per il posto occupato da *A Logical Calculus of the Ideas Immanent in Nervous Activity* nella storia del pensiero computazionale e delle scienze cognitive si rimanda alla monumentale opera di Margaret Boden, *Minds as Machine* del 2006 pp.182-195; Abraham, 2002 e ad Arbib, 2000. I rapporti tra la cibernetica wieneriana e le idee di McCulloch e Pitts sono ricostruite in Masani, 1990b.

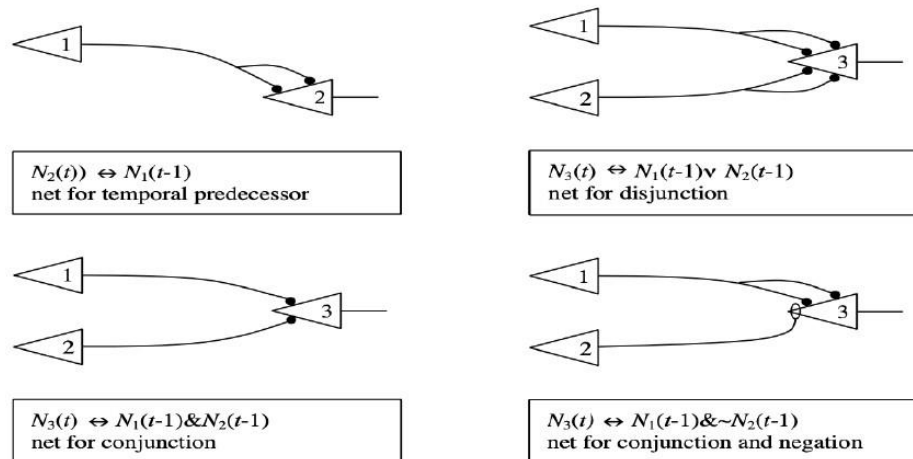


Fig.8: I connettivi logici di (1) *precession*; (2) *disjunction*; (3) *conjunction* e (4) *conjoined negation* di McCulloch, Pitts, 1943 in Piccinini, 2004 [Per esigenze grafiche si è qui sostituita la notazione dei *Principia Mathematica* di Russell e Whitehead [1910-1913] usata dai due autori con quella della logica proposizionale contemporanea] In senso orario, le reti neurali illustrate rappresentano quattro operazioni logiche fondamentali. La prima rete (in alto a sinistra) rappresenta la (1) precedenza temporale (identità), in cui il neurone $N_2(t)$ si attiva in corrispondenza dell'attivazione precedente ($t - 1$) di N_1 descritta dalla formula logica $N_2(t) \leftrightarrow N_1(t - 1)$. La seconda rete (in alto a destra) implementa la (2) disgiunzione logica (OR), attivando $N_3(t)$ quando almeno uno tra $N_1(t - 1)$ o $N_2(t - 1)$ è attivo, ed è descritta dalla formula $N_3(t) \leftrightarrow N_1(t - 1) \vee N_2(t - 1)$. La terza rete (in basso a destra) combina una (3) congiunzione e una negazione (AND-NOT): $N_3(t)$ si attiva solo se $N_1(t - 1)$ è attivo e $N_2(t - 1)$ è inattivo e corrisponde a $N_3(t) \leftrightarrow N_1(t - 1) \wedge \neg N_2(t - 1)$. Infine, la quarta rete (in basso a sinistra) rappresenta la (4) congiunzione logica (AND), in cui $N_3(t)$ si attiva solo se sia $N_1(t - 1)$ che $N_2(t - 1)$ sono attivi ed è espressa da $N_3(t) \leftrightarrow N_1(t - 1) \wedge N_2(t - 1)$. Queste configurazioni dimostrano come le reti McCulloch-Pitts possano modellare operazioni logiche di base con precisione.

neuroni del cervello, per tirare i fili della nostra analisi, non sono altro, secondo questo modello, che una delle molteplici possibili realizzazioni - al pari di relè e circuiti - di proposizioni logiche. La portata esplicativa di tale modello, e la sua logica, nell'analogia tra calcolatore e cervello sarà più volte oggetto di esame e costituirà uno degli snodi interpretativi nello sviluppo e nel design dei calcolatori, così come nel rapporto di questi con l'essere umano, entrambi concepiti come sistemi di *information processing*.

Il cervello, malgrado la sua incredibile complessità organica, può, secondo questo punto di vista, essere, a diversi livelli d'astrazione, modellizzato¹⁵³ come una macchina digitale.

¹⁵³ Si rimanda qui alle considerazioni intorno alla black box e alla modellizzazione.

Vedremo come l'approfondimento del suo eventuale carattere puramente digitale comporterà una "riforma" della logica proposizionale binaria proposta nel calcolo neuronale del modello di McCulloch e Pitts.

McCulloch, nel suo articolo scritto con John Pfeiffer *On Digital Computers Called Brains* del 1949 [McCulloch, Pfeiffer, 1949], descrive il cervello come una macchina digitale composta da unità discrete, operanti secondo le modalità già delineate in *A Logical Calculus* del 1943, e attribuisce al lavoro di Shannon [1939] il primato di aver "applicato l'algebra booleana alle reti di relè" [McCulloch, Pfeiffer, 1949, trad. mia]¹⁵⁴. Tale passaggio, come abbiamo visto, si fonda sulla capacità binaria – comune tanto ai neuroni quanto ai circuiti artificiali – equivalente allo 0/1 dei bit, divenuti misura universale delle capacità informative dei sistemi, siano essi biologici o artificiali.

Proprio i *bit* vengono da McCulloch visti come un decisivo sviluppo della scienza delle comunicazioni, in quanto rappresentano quelle "unità ultime" [McCulloch, Pfeiffer, 1949, trad. mia]¹⁵⁵, al pari dei geni nella biologia o agli atomi nella fisica e nella chimica, decisivi per lo sviluppo epistemologico di una materia [McCulloch, Pfeiffer, 1949, trad. mia]¹⁵⁶.

Shannon aveva compiuto un passo decisivo nell'individuare la struttura matematica di quelli che, nello studio sulla ridondanza, egli definiva gli "atomi della comunicazione", ossia le lettere. Come osserva James Gleick, nell'analisi sull'occorrenza probabilistica delle lettere di un linguaggio naturale - necessaria alla quantificazione dell'informazione come entropia - Shannon aveva elaborato un concetto molto vicino agli "atomi linguistici" di Edward Sapir (1884–1939). Il linguista polacco scriveva nel 1921: "I meri suoni del linguaggio non costituiscono la parte essenziale della lingua. Questa parte essenziale deve essere ricercata piuttosto nella classificazione, nella strutturazione formale. [...] Ancora una volta la lingua, come struttura, costituisce, nel suo aspetto interno, lo stampo del pensiero" [Sapir, 1921, p.22 e p.56 in Gleick, 2011, pp.199-200 trad. it]¹⁵⁷.

¹⁵⁴ "So far as we can discover, Shannon was the first to apply Boolean algebra to nets of relays".

¹⁵⁵ "Ultimate units".

¹⁵⁶ "The art of communication is very old, but the science is very new. When we reduced our speech to syllables and built an alphabet, we took the first great steps in the right direction, for our letters are comparable to the atoms of chemistry. All communication is a matter of information conveyed by signals, and the next great step was atomic signals. [...] There is another kind of signal in which some variable is restricted to a small number of possible values at distinct times or separate places. The message is the order or number of these possible quantities, as in the case of the dots and dashes of telegraphy and the coin of the realm. In the simplest case, this reduces to the presence or absence-call it one or zero-of something at an appointed place or time. This is the famous binary digit, now nicknamed "bit".

¹⁵⁷ L'esposizione del 1950 di Shannon alle *Macy Conferences*, presentata di fronte a una platea composta dai protagonisti del movimento cibernetico, rappresenta una prova tangibile dell'impatto che le sue teorie hanno avuto ben oltre il campo dell'ingegneria delle comunicazioni. Esse hanno influenzato profondamente la

Da questa prospettiva il pensiero ha una struttura probabilistica quantificabile con il trattamento matematico elaborato da Shannon, con strumenti che sono direttamente riconducibili alla loro origine nel contesto bellico. Così come Wiener era giunto alla sua *weltanschauung* cibernetica dal lavoro sul predittore antiaereo, Shannon era arrivato alla sua teoria dell'informazione, come si è cercato di dimostrare qui, dal suo lavoro crittoanalitico per il progetto X. "L'industria artigianale dei nomi in codice informatici-ENIGMA, ENIAC, MANIAC,

matematica, la neuropsicologia, le scienze della cognizione e altri campi del sapere che costituiscono il complesso e multidisciplinare insieme disciplinare della cibernetica. Durante l'evento si sviluppò un vivace dibattito tra i presenti. Un momento particolarmente significativo fu la domanda fatta da Walter Pitts, che pose la questione a Shannon sull'applicabilità delle sue teorie a lingue dalla struttura diversa rispetto all'inglese, come il cinese [Shannon, 1950, p.262]. Questa lingua, basata su ideogrammi anziché su un alfabeto fonetico, sollevava questioni sull'universalità della teoria dell'informazione e ne sfidava l'applicabilità a sistemi linguistici profondamente differenti. Una prova indiretta della pertinenza della questione del nesso pensiero-linguaggio-informazione si può ritrovare nell'enfasi posta da Doug Engelbart nel suo *Augmenting Human Intellect: A Conceptual Framework* [Engelbart,1962], in cui il "matching" tra uomo e macchina, finalizzato a potenziare le capacità intellettive e corporee dell'utente umano nel suo interfacciarsi con la tecnologia, si basa sulla teoria di Edward Sapir e del suo allievo e collega Benjamin Lee Whorf, (1897–1941) nota come Ipotesi di Sapir-Whorf (*Sapir-Whorf Hypothesis*, SWH) della relatività linguistica. Questa teoria afferma che il linguaggio che parliamo influisce significativamente sul nostro modo di pensare e percepire la realtà: "Il sistema linguistico di sfondo (in altre parole la grammatica) di ciascuna lingua non è soltanto uno strumento di riproduzione per esprimere idee, ma esso stesso dà forma alle idee, è il programma e la guida dell'attività mentale dell'individuo, dell'analisi delle sue impressioni, della sintesi degli oggetti mentali di cui si occupa" [Whorf, 1956, p.xi-xii trad. it.]. Per Engelbart, ciò che permetteva alle macchine digitali e ai computer di aumentare le capacità dell'essere umano era il fatto che per questi "artefatti" i processi che collegano l'essere umano e la macchina al mondo esterno, e tra di loro, potevano essere trovati nel linguaggio naturale, inteso sia nelle sue dimensioni fisiologiche che sociali, un linguaggio accessibile a tutti. Tra tutti gli strumenti usati dall'uomo per costruire artefatti capaci di modificare l'ambiente e sé stessi: "il linguaggio appariva chiaramente come lo strumento-meta, quello che rendeva possibili tutti gli altri" ["Of all the tools humans use to construct and modify the machines they use to transform themselves and their environment, language clearly seemed the meta-tool, the one that made all the others possible. Bardini, 2000, p. 3e, trad. it. mia]. Il computer diventava lo strumento privilegiato di manipolazione dei simboli. Engelbart scrive che: "la visualizzazione computerizzata di nuovi simboli dovrebbe quindi consentirci di influenzare il modo in cui concettualizziamo il nostro mondo. Il computer potrebbe così diventare un mezzo aperto utilizzabile per 'dare senso al mondo', per mappare la struttura del mondo come flusso di informazioni al fine di gestirne la complessità crescente" ["The computer thus could become an open medium that could be used to "make sense of the world," to map the structure. of the world as information flows in order to manage their increasing complexity. Bardini, 2000, p. 37, trad. it. mia]. Questo concetto si basa sulla recezione della SWH. Engelbart stesso afferma: "L'Ipotesi Whorfiana afferma che 'la visione del mondo di una cultura è limitata dalla struttura della lingua che tale cultura utilizza'. Tuttavia, sembra esserci un altro fattore da considerare nell'evoluzione del linguaggio e della capacità di ragionamento umano. Proponiamo la seguente ipotesi, correlata all'ipotesi Whorfiana: sia la lingua utilizzata da una cultura, sia la capacità di un'attività intellettuale efficace, sono direttamente influenzate durante l'evoluzione dai mezzi con cui gli individui controllano la manipolazione esterna dei simboli" ["The Whorfian hypothesis states that the world view of a culture is limited by the structure of the language which that culture uses. But there seems to be another factor to consider in the evolution of language and human reasoning ability. We offer the following hypothesis, which is related to the Whorfian hypothesis: both the language used by a culture and the capability for effective intellectual activity are directly affected, during their evolution, by the means by which individuals control the external manipulation of symbols. For identification, we will refer to this as the Neo-Whorfian hypothesis", Engelbart 1962, p. 24, trad. it. mia]. L'interazione tra calcolatore ed essere umano in termini linguistici nasce qui. Non può non saltare agli occhi, inoltre, l'analogia concettuale con quanto Shannon aveva realizzato nell'analisi simbolica dei circuiti [Shannon, 1939]. La rappresentazione mediante l'algebra, ossia attraverso un sistema strutturato di simboli, permetteva di semplificare la complessità dei circuiti. La differenza risiede nel fatto che ora questa visualizzazione simbolica viene applicata alle strutture (allo *stampo* e agli *atomi* del pensiero stesso).

TUNNY, BOMBE, COLOSSUS, SIGSALY, e così via-rivelava una richiesta di potenza per la decodifica che stava alimentando una rivoluzione nell'informatica, guidata da segrete burocrazie dell'intelligence" [Soni, Goodman, 2017, p. 96 trad. mia]¹⁵⁸.

Il pensiero dunque ha uno *stampo* stocastico informazionale e il cervello¹⁵⁹ è un sistema *negentropico* a feedback che, al pari di ogni altro sistema organico o artificiale, realizza uno scambio di informazioni con l'ambiente.

Abbiamo ora gli strumenti - prima che la nostra analisi si approfondisca prendendo in esame come tutto ciò si sia concretizzato a livello di hardware dei calcolatori - per tornare alla domanda iniziale: come ha fatto un pensiero, forgiato nel contesto bellico e ricostruito qui in alcune delle sue tappe fondamentali, fulcro della "retorica cibernetica della ricerca mainstream militare-industriale-accademica" [Turner, 2006, p. 33 trad. mia]¹⁶⁰ a essere riletto e considerato come uno strumento di liberazione individuale e collettiva, in armonia con l'ambiente?

¹⁵⁸ "The cottage industry of computer code names-ENIGMA, ENIAC, MANIAC, TUNNY, BOMBE, COLOSSUS, SIGSALY, and so on-evinced a demand for codebreaking power that was spurring a revolution in computing, presided over by secretive intelligence bureaucracies".

¹⁵⁹ A questo proposito, anche con lo scopo di segnare la differenza tra le posizioni cibernetiche e informazionali e il quelle del meccanicismo materialista, che, a mio avviso, corre in parallelo con lo scarto tra la cibernetica stessa e il comportamentismo psicologico e in filosofia della mente, passo celebre da *Cybernetics* di Wiener: "Il cervello meccanico non secerne pensiero «come il fegato la bile», come pretendevano i vecchi materialisti, né lo produce sotto forma di energia, come fanno i muscoli per la loro attività. L'informazione è informazione, non materia o energia. Al giorno d'oggi, nessun materialismo che non ammetta questo può sopravvivere" [Wiener, 1950, p.177 trad.it.]. Il ruolo del feedback negativo nei sistemi, come si è analizzato *supra* introducendo il tema del machine learning, è stato preso in esame anche da Shannon in *Computers and Automata* [Shannon, 1953] in cui prende in esame le macchine computazionali non numeriche e loro caratteristiche e della possibilità di queste, in confronto al cervello umano di giocare ed apprendere. A questo proposito Shannon da una sua definizione di apprendimento: "Il concetto di apprendimento, come quelli di pensiero, coscienza e altri termini psicologici, è difficile da definirsi in modo preciso e accettabile dalle varie parti interessate. Una formulazione approssimativa potrebbe essere la seguente. Supponiamo che un organismo o una macchina possano essere collocati o connessi con una classe di ambienti, e che esista una misura della "riuscita" o "adattamento" all'ambiente. Supponiamo inoltre che questa misura sia comparativamente ristretta nel tempo, cioè che si possa misurare la riuscita su periodi di tempo brevi rispetto alla durata della vita dell'organismo. Se il valore risultante di questa misura ristretta di riuscita tende ad aumentare, col passare del tempo, per la classe di ambienti in questione, potremmo allora dire che l'organismo o la macchina imparano ad adattarsi a tali ambienti relativamente alla misura di riuscita che è stata scelta. L'apprendimento acquista un significato quantitativo in termini di ampiezza e complessità della classe di ambienti cui la macchina può adattarsi." [Shannon, 1953, p.101 trad. it.]. Lo stesso Shannon, noto per la sua passione nella realizzazione di macchinari, aveva scritto un articolo sulla possibilità di programmare un computer per giocare a scacchi [1950b] e realizzato un automa "capace di apprendere", Il "Topo di Shannon" è un dispositivo elettromeccanico progettato per risolvere un labirinto configurabile in una griglia 5x5. Un piccolo topo magnetico, guidato da un sistema di relè con circa 110 componenti, esplora il labirinto tramite tentativi ed errori, memorizzando i percorsi corretti. Una volta esplorato, è in grado di raggiungere la "scatola del cibo" direttamente da qualsiasi punto già visitato senza errori. Se il labirinto viene modificato, il topo prova percorsi alternativi aggiornando la sua memoria, mostrando la capacità di apprendere e adattarsi [Shannon, 1953]. Per la storia, e soprattutto per la questione epistemologica sottesa, degli automi e di entità *artificiali* che implementano istanze psicologiche o fisiologiche ascrivibili all'essere umano o a organismi biologici sono imprescindibili Cordeschi, 1994 e 1998.

¹⁶⁰ "The cybernetic rhetoric of mainstream military-industrial-academic research".

La cibernetica si diffuse anche grazie al ruolo di alcune figure-ponte tra il mondo tecnologico e quello controculturale, come Stewart Brand (1938–vivente). Brand è noto per aver lanciato, nel 1968, il *Whole Earth Catalog* [Brand, ed. 1968]¹⁶¹ una pubblicazione ideata come guida per i "*back-to-the-landers*" e la controcultura dell'epoca. Il *Catalog* fungeva da piattaforma informativa che metteva a disposizione strumenti, tecnologie e risorse per una vita autosufficiente, combinando materiali pratici con riflessioni sulla tecnologia e sull'ecologia. Era ideato come un network editoriale capace di connettere comunità rurali, pensatori tecnologici e ambientalisti. Brand seppe integrare la critica del *New Communalism* alla politica gerarchica con la teoria dei sistemi, trovando in quest'ultima una metodologia editoriale e gestionale adatta a coordinare comunità molteplici

Questa prospettiva, intrecciata alla retorica di frontiera dei "cowboy e indiani," offriva ai lettori del *Whole Earth Catalog* un modo per collegare i loro tentativi controculturali di trasformare sé stessi e le proprie comunità al mito americano. Come il pubblico del *New Communalism*, il *Catalog* celebrava le tecnologie su piccola scala come strumenti per migliorare la vita individuale. Tuttavia, proponeva anche queste tecnologie - e sé stesso - come prototipi di una nuova relazione tra l'individuo, l'informazione e la tecnologia.

In questo senso, il *Whole Earth Catalog* non era soltanto una raccolta di risorse, ma un manifesto culturale che proponeva una visione trasformativa della società. Secondo la cibernetica, la mente umana possiede una struttura informazionale: ciò implica che le tecnologie computazionali, eredi della cibernetica sviluppata per scopi bellici, potessero essere utilizzate per ampliare, potenziare e liberare la mente stessa. In questo contesto, tali tecnologie erano concepite come l'equivalente dell'LSD, delle droghe psichedeliche e delle luci stroboscopiche nei concerti rock, strumenti capaci di espandere la coscienza e trasformare la percezione.

Non sorprende, quindi, che tra le varie sezioni del *Whole Earth Catalog* - che nel 1972 vinse il *National Book Award* - la prima fosse intitolata *Understanding Whole Systems*"¹⁶² che nel

¹⁶¹ Per la vicenda di Brand, del *Whole Earth Catalog* e dell'influenza controculturale sulla *digital culture* contemporanea si rimanda a Turner, 2006.

¹⁶² La prima edizione del *Catalog* di 61 pagine, fu finanziata con eredità personali da Brand e dalla moglie e conteneva 133 articoli selezionati da Brand stesso. Venduto a cinque dollari, aveva in copertina una foto della Terra vista dallo spazio con il motto "*WHOLE EARTH CATALOG / access to tools*". Nel 1971, l'anno in cui vinse *National Book Award*, il *Catalog* raggiunse 448 pagine, vendette oltre un milione di copie. Era articolato in sette sezioni. Oltre alla già citata *Understanding Whole Systems*, dedicata alla comprensione dei sistemi complessi, sia sociali che ecologici, in linea con la teoria dei sistemi cibernetici, le altre erano *Shelter and Land Use*, dedicata a fornire risorse per la costruzione e la gestione di abitazioni e terreni, con un'attenzione particolare alla sostenibilità e all'autosufficienza; *Industry and Craft* che si concentrava su tecniche artigianali e strumenti per la produzione indipendente, promuovendo il ritorno a competenze pratiche e manuali; *Communications* che offriva

catalogo comparisse in evidenza *Cybernetics* di Wiener. Ogni sistema, infatti, è inserito in un altro sistema con cui scambia informazioni, perseguendo omeostaticamente una condizione di equilibrio. Brand, e con lui il movimento controculturale - di cui la poesia di Brautigan citata in apertura costituisce un esempio - rispose, attraverso la cibernetica, "alla politica gerarchica con la teoria dei sistemi" [Turner, 2006, p.99 trad. mia]¹⁶³. Nell'ottica del cambiamento sociale, "agendo su piccola scala, l'individuo può imitare il tiratore scelto di Norbert Wiener, aggiustando il suo tiro" [Turner, 2006, p.100 trad. mia]¹⁶⁴

. Questo il *milieu* che influenzerà la nascita della computazione personalizzata favorita, come vedremo dagli sviluppi dell'elettronica a stato solido - come analoghi funzionali dei neuroni disponibili e a basso costo - e da vicende imprenditoriali nate "dal basso", come l'Homebrew Computer Club favorite dalle dinamiche culturali e tecnologiche in corso, in opposizione alla retorica veicolata da colossi come l'IBM. [Rheingold, 1985, Markoff, 2005].

Nel dicembre 1972, Stewart Brand pubblicò un articolo su *Rolling Stone* intitolato *Spacewar: Fanatic Life and Symbolic Death Among the Computer Bums*, in viene descritto l'avvento computer come "la miglior cosa dagli psichedelici". Brand osservava come i computer, inizialmente sviluppati nell'ambito della ricerca bellica, avessero il potenziale per trasformarsi in strumenti di liberazione e potenziamento mentale, in sintonia con le visioni romantiche di pionieri come Norbert Wiener, Warren McCulloch, J.C.R. Licklider, John von Neumann e Vannevar Bush¹⁶⁵.

tecnologie e risorse per migliorare la comunicazione, includendo le prime tecnologie digitali emergenti, anticipando così il futuro della connettività; *Community* che proponeva idee e strumenti per costruire e sostenere comunità resilienti e cooperative.; *Nomadics* si rivolgeva a chi adottava uno stile di vita nomade, con risorse per viaggiare e vivere in modo sostenibile; *Learning* metteva a disposizione libri e strumenti per l'apprendimento autonomo, promuovendo la crescita personale e culturale degli individui.

¹⁶³ "Brand's response resonates with the countercultural critique of hierarchical politics and with elements of systems theory".

¹⁶⁴ "By acting on a small scale, the individual can imitate Norbert Wiener's gunner, adjusting his fire".

¹⁶⁵ "Ready or not, computers are coming to the people. That's good news, maybe the best since psychedelics. It's way off the track of the "Computers - Threat or menace?" school of liberal criticism but surprisingly in line with the romantic fantasies of the forefathers of the science such as Norbert Wiener, Warren McCulloch, J.C.R. Licklider, John von Neumann and Vannevar Bush". L'articolo di Brand inoltre fa riferimento agli "hackers" definiti - intervistando Alan Kay (1940-vivente) figura cardine prima dell'*Augmentation Research Center* (ARC) diretto da Engelbert, situato presso lo *Stanford Research Institute* (SRI) dello *Xerox Lab* di Palo Alto e dello sviluppo del personal computer- come me "fanatici" della tecnologia, brillanti e ossessionati dal spingere i limiti del possibile con i computer, trasformando la programmazione in un'arte creativa e accessibile. Nell'articolo si fa anche riferimento ad ARPANET, una rete di computer sviluppata negli anni '60 dall'*Advanced Research Projects Agency* (ARPA) del Dipartimento della Difesa degli Stati Uniti per collegare università e centri di ricerca, l'antenata dell'odierna Internet. Per il rapporto tra controcultura e i protagonisti dell'Intelligenza Artificiale e della *Human-Computer Interaction*, che indaga le tecnologie come strumenti "*mind expanding*", si veda Rheingold, 1985. Il rapporto controcultura e sviluppo della computazione personale è indagato anche in Markoff, 2005. L'hacker come "eroe" dell'era dell'informazione portatore di una propria etica e una propria cultura, insieme alla descrizione dei protagonisti del fenomeno è in Levy, 1984.

Dopo aver esaminato le condizioni scientifiche, culturali (e contro-culturali) che hanno accompagnato la nascita della cibernetica e favorito lo sfumarsi delle differenze tra esseri umani e macchine, entrambe concepite come sistemi di elaborazione dell'informazione, matematicamente quantificabile in modo rigoroso, è ora necessario spostare l'analisi a un livello più alto di astrazione. In questo contesto occorre interrogarsi sulle capacità e sui limiti sia dei sistemi artificiali sia delle loro controparti biologiche. Un passaggio imprescindibile in questa direzione è rappresentato dal lavoro di Alan Turing, che costituisce un punto genetico fondamentale nello sviluppo delle nuove macchine e apre prospettive decisive sulle potenzialità e sulle relazioni tra calcolatore, cervello e mente umana.



Fig.9 *Cybernetics* di Norbert Wiener nella prima edizione del *Whole Earth Catalog* dell'Autunno 1968 [Brand ed, 1968]

Cap. 2: Dalla carne al calcolo e ritorno: Computazioni; Imitazioni; Implementazioni.

1. Formalità: *Ignorabimus quia Incompletum*

Il percorso che abbiamo delineato nelle pagine precedenti, profondamente influenzato dalle vicende belliche, si strutturava intorno all'analogia formale tra il cervello e il calcolatore, entrambi concepiti come sistemi di elaborazione computazionale dell'informazione, calcolata nella forma matematicamente rigorosa definita da Shannon. È necessario, a questo punto, risalire alle origini di quella che potremmo chiamare la *metafora cibernetica*, ricostruendone la genesi concettuale e mostrando come essa abbia condotto alla nascita delle macchine di calcolo. Queste, come vedremo, hanno aperto nuove prospettive sul loro corrispettivo biologico, il cervello umano, mostrando capacità e limiti di entrambi nell'elaborazione computazionale dell'informazione. Queste riflessioni - che trovano il loro apice nel lavoro di Alan Turing e nelle sue conseguenze - e che, passando attraverso la crisi dei fondamenti matematici, ci hanno portato fino all'assoluta California dei *new communalists*, nascono nel cuore dell'Europa.

Parigi, nell'agosto del 1900, era avvolta dall'afa. In queste giornate roventi, dal 6 al 12 del mese, si tenne un evento intellettuale che avrebbe, al pari del sole che scaldava la città, illuminato-anzi, infuocato-l'orizzonte scientifico e intellettuale mondiale per gli anni a venire. Si trattava del Congresso Internazionale dei Matematici (ICM), evento nato sotto gli auspici del grido di battaglia lanciato dal tedesco Felix Klein (1849-1925) a Chicago nel 1893: "Matematici di tutto il mondo, unitevi!"¹⁶⁶

Quella di Parigi era la seconda edizione del congresso, dopo l'incontro inaugurale tenutosi a Zurigo nel 1897. L'8 agosto si attendeva con particolare trepidazione il discorso del leggendario David Hilbert (1862-1943), che solo l'anno prima aveva pubblicato i *Grundlagen der Geometrie* [Hilbert, 1899], "il lavoro più rilevante in materia dai tempi di Euclide stesso" [Berto, 2009, p.46]. Proveniente dall'Università di Gottinga-la capitale mondiale della matematica dell'epoca-Hilbert, con l'ottimismo scientifico che lo contraddistingueva¹⁶⁷,

¹⁶⁶ Per una storia dell'ICM, che continua a svolgersi ancora oggi (l'edizione del 2022, prevista a Helsinki, si è tenuta online a causa della pandemia da COVID-19, mentre la prossima edizione si terrà a Philadelphia nel 2026), si vedano Curbera (2009) e Lehto (1998).

¹⁶⁷ Lo spirito scientifico ottimista di Hilbert, riflesso in tutto il suo programma di ricerca, è immortalato nella celebre iscrizione sulla sua lapide: "*Wir müssen wissen, wir werden wissen*" ("Dobbiamo sapere, sapremo"). Per la

presentò una lista di 23 problemi [Hilbert, 1902] destinati a tracciare la rotta della ricerca matematica nel secolo che stava per aprirsi.

Il secondo di questo elenco di problemi è particolarmente significativo in quanto contiene *in nuce* il programma con cui Hilbert vorrà dare risposta alla cosiddetta “crisi dei fondamenti” in matematica, aperta con la scoperta delle antinomie della teoria insiemistica cantoriana¹⁶⁸ e “ufficializzata” con la scoperta, da parte di Bertrand Russell del paradosso indicato nella lettera di quest’ultimo del 16 giugno 1902 a Gottlob Frege [Russell, lettera a Frege 16 giugno 1902]¹⁶⁹.

vicenda biografica e scientifica del matematico e il suo rapporto con lo sviluppo della matematica nel Novecento, si vedano Lolli, 2011 e 2016. Per la vicenda intellettuale di Hilbert in relazione al pensiero computazionale, si veda Davis, 2000.

¹⁶⁸ Georg Cantor (1845-1918), matematico tedesco, espose le sue idee sugli insiemi nel saggio *Über eine Eigenschaft des Inbegriffes aller reellen algebraischen Zahlen* (1874), dove dimostrò che l’insieme dei numeri reali è più grande di quello dei numeri naturali. Questo risultato fu ottenuto attraverso il metodo della *diagonalizzazione*, che mostrava come non fosse possibile costruire una corrispondenza biunivoca tra numeri naturali e numeri reali, stabilendo così la distinzione tra insiemi numerabili e non numerabili. Per descrivere questi nuovi tipi di infinito, sviluppò la teoria dei numeri transfiniti, introducendo il concetto di cardinalità dei numeri $\aleph$ (aleph) per classificare insiemi infiniti di grandezze diverse. Tuttavia, la scoperta di paradossi insiemistici, legati all’insieme di tutti gli insiemi e all’insieme potenza, mise in crisi la sua teoria. Come osservò il filosofo statunitense W.V.O. Quine, proprio in virtù di questi paradossi, la teoria insiemistica originaria si rivelò essere «un lupo travestito da agnello» [Berto, 2009, p.7]. Per risolvere queste difficoltà, furono sviluppate nuove teorie, tra cui la teoria assiomatica di Zermelo-Fraenkel, che fornì basi più rigorose all’insiemistica. Si vedano Berto, 2009, pp.44-45; Davis, 2000, pp.83-108 trad. it..

¹⁶⁹ Gottlob Frege (1848-1925) filosofo e matematico tedesco, è considerato tra i fondatori della logica moderna. Con il suo *Begriffsschrift* del 1879, introdusse un linguaggio simbolico rigoroso per la logica, gettando le basi della logica predicativa. Nel *Die Grundlagen der Arithmetik* [Frege, 1884], sviluppò l’idea che i concetti matematici potessero essere definiti esclusivamente in termini logici, un progetto che culminò nei *Grundgesetze der Arithmetik* (1893-1903). Inoltre, nel celebre esempio della “stella del mattino” e della “stella della sera”, presente nel saggio *Über Sinn und Bedeutung* del 1892 [Frege, 1982], definì la distinzione tra senso e riferimento, contribuendo in modo decisivo alla filosofia del linguaggio. Il cuore del *programma logicista* di Frege era l’idea che l’aritmetica derivasse interamente dalla logica, senza alcun bisogno di appoggiarsi a intuizioni psicologiche, costruendo tutto su principi puramente formali. Nella famosa lettera a lui indirizzata, Russell, pur esprimendo rispetto e ammirazione per il suo programma, gli comunica una difficoltà insormontabile che mina le basi del *logicismo*. Scopre infatti un paradosso che emerge quando si considerano insiemi definiti in base a proprietà autoapplicative, scrivendo: “Un tempo lo credevo anch’io, ma ora questo punto mi sembra dubbio a causa della seguente contraddizione. Sia w il predicato: ‘un predicato che non può essere predicato di sé stesso.’ w può essere predicato di sé stesso? Qualunque risposta si scelga, ne segue l’opposto. Dunque, dobbiamo concludere che w non è un predicato. Analogamente, non esiste una classe (intesa come totalità) di quelle classi che, prese come totalità, non appartengono a sé stesse” [“This I formerly believed, but now this view seems doubtful to me because of the following contradiction. Let w be the predicate: ‘a predicate that cannot be predicated of itself.’ Can w be predicated of itself? From each answer its opposite follows. Therefore we must conclude that w is not a predicate. Likewise there is no class (as a totality) of those classes which, each taken as a totality, do not belong to themselves.” Russell, 16 giugno 1902 lettera a Frege trad. mia]. Questa scoperta segna l’inizio della *crisi dei fondamenti* della matematica, mettendo in dubbio la coerenza del *programma logicista* di Frege e aprendo la strada ai successivi sviluppi della logica e della teoria degli insiemi.

L'antinomia in questione è riassumibile nella proprietà di non appartenere a sé stessi.

Formalizzata, essa è esprimibile come: posto $R = \{x \mid x \notin x\}$ ¹⁷⁰, se supponiamo che $R \in R$,

¹⁷⁰ Il paradosso di Russell può essere formulato come: "L'insieme di tutti gli insiemi che non appartengono a sé stessi appartiene a sé stesso se e solo se non appartiene a sé stesso". A questa struttura appartengono altri paradossi celebri, come quello del barbiere, che afferma: "In un villaggio vi è un solo barbiere, un uomo ben sbarbato, che rade tutti e solo gli uomini del villaggio che non si radono da soli. La domanda è: il barbiere si fa la barba da solo?". Il più famoso, e probabilmente il più antico di tutti, è il paradosso del mentitore, attribuito a Epimenide di Creta, che dichiarò: "Tutti i Cretesi sono bugiardi". L'affermazione generava una contraddizione, in quanto, se vera, significherebbe che chi la pronuncia sta anch'egli mentendo. Tutti questi paradossi sono riducibili alla forma autoriferita "Questo enunciato è falso". Secondo Russell, come evidenziato nella sua lettera a Frege del giugno 1902, il problema di questi enunciati è proprio l'autoreferenzialità, che genera contraddizioni insostenibili all'interno di un sistema formale. Per eliminare questo vizio, Russell sviluppò la *teoria dei tipi*, che mirava a escludere le costruzioni autoriferite e prevenire simili paradossi logici. Per Russell e Whitehead, i paradossi derivano da "un certo tipo di circolo vizioso. Questi circoli viziosi nascono dall'ipotesi che un insieme di oggetti possa contenere membri che possono essere definiti solo tramite il riferimento all'insieme stesso considerato nella sua totalità." [An analysis of the paradoxes to be avoided shows that they all result from a certain kind of vicious circle. The vicious circles in question arise from supposing that a collection of objects may contain members which can only be defined by means of the collection as a whole", Russell, Whitehead 1910-13, p.37, trad. it. mia]. Il principio della totalità, per i due studiosi inglesi, stabilisce che un insieme non può contenere elementi che possono essere definiti solo facendo riferimento all'insieme stesso. Questo evita i circoli viziosi nei quali una definizione dipende da ciò che sta definendo, causando così le contraddizioni. "Il principio che ci permette di evitare le totalità illegittime può essere formulato come segue: 'Ciò che coinvolge tutti i membri di un insieme non può essere uno dei membri di quell'insieme', oppure, in modo equivalente: 'Se un insieme, supposto come totalità, contiene membri definibili solo in termini di questa totalità, allora tale insieme non può costituire una totalità ben definita'" ["The principle which enables us to avoid illegitimate totalities may be stated as follows: "Whatever involves all of a collection must not be one of the collection"; or, conversely: "If, provided a certain collection had a total, it would have members only definable in terms of that total, then the said collection has no total", Russell, Whitehead 1910-13, p.37, trad. it. mia]. A partire da questo, la soluzione proposta nella teoria dei tipi consisteva nell'assegnare a ogni entità dell'universo un tipo, ad es.:

- Tipo 1: individui (entità che non sono insiemi, come una mucca).
- Tipo 2: insiemi i cui membri sono solo di tipo 1.
- Tipo 3: insiemi i cui membri sono solo di tipo 2.

Gli insiemi contenenti membri di più tipi erano vietati, così come ogni affermazione del tipo $x \in x$ o $x \notin x$ eliminando così contraddizioni come (1) e (2) [Masani, 1990 pp.48-53]. Il pensiero di Russell non rappresenta soltanto un nodo fondamentale nel processo di assiomatizzazione e formalizzazione logica e matematica del pensiero, da cui prenderà forma la computazione contemporanea, ma ha svolto anche un ruolo decisivo nello sviluppo della cibernetica. Wiener, grazie all'interessamento del padre Leo, che scrisse all'eminente logico britannico, svolse il dottorato sotto la guida di Russell, il quale tuttavia non rimase impressionato né da lui né dal padre [Masani, 1990, pp.45-48]. In seguito, Wiener si rimproverò di aver trascurato del tutto la portata filosofica della teoria dei tipi [Masani, 1990, p.48], che nondimeno esercitò una notevole influenza sul pensiero del futuro padre della cibernetica [Masani, 1990, pp.45-66]. Un'altra figura centrale del pensiero cibernetico, Gregory Bateson, pose la teoria russelliana dei tipi al centro della sua riflessione. Bateson osservò che, nella comunicazione umana, la confusione tra diversi livelli logici può generare effetti patologici. Nella schizofrenia, ad esempio, l'incapacità di distinguere tra il contenuto di un messaggio e il suo metamessaggio porta a situazioni di doppio vincolo, in cui il soggetto è intrappolato in contraddizioni comunicative irrisolvibili. Nella teoria dell'apprendimento, Bateson osserva che questo non si riduce a una mera acquisizione di informazioni, ma implica il riconoscimento del livello logico in cui esso avviene. L'influenza della teoria dei tipi è esplicita nel passaggio dall'*imparare* a quello dell'*imparare ad imparare*, un principio che ha importanti conseguenze nella riflessione sulla cibernetica e sulla teoria dei sistemi. Questa stessa dinamica si manifesta nel rischio di errori epistemologici, e cioè nell'affrontare problemi sistemici con soluzioni di livello inadeguato. Per questo motivo, in tutti questi

allora per la definizione di R deve valere $R \notin R$, il che è una contraddizione. Se invece supponiamo che $R \notin R$, allora per la stessa definizione deve valere $R \in R$, altra contraddizione. Ne segue che entrambe portano a $R \in R \Leftrightarrow R \notin R$. Il che è assurdo.

Nella formulazione di questo secondo problema, quello della *coerenza*, Hilbert afferma:

Ma soprattutto desidero designare la seguente come la questione più importante tra le numerose domande che si possono porre sugli assiomi: dimostrare che non sono contraddittori, cioè che un numero finito di passi logici basati su di essi non può mai condurre a risultati contraddittori [...] Per mostrare l'importanza del problema da un altro punto di vista, aggiungo la seguente osservazione: se vengono assegnate a un concetto proprietà contraddittorie, dico che matematicamente il concetto non esiste. [...] invece un sistema di entità le cui relazioni reciproche sono governate dagli assiomi stabiliti e per il quale sono vere tutte e solo le proposizioni che possono essere derivate dagli assiomi attraverso un numero finito di processi logici [Hilbert, 1902]¹⁷¹.

In queste parole del matematico di Gottinga si trova già la sua soluzione alla crisi che la matematica stava attraversando: il formalismo:

La posizione secondo cui la matematica non è un insieme di proposizioni che rappresentano un settore astratto della realtà, ma è molto più simile a un gioco, senza comportare un impegno ontologico verso oggetti o proprietà più di quanto normalmente si pensi che accada giocando a ludo o a scacchi [Weir, 2024]¹⁷².

ambiti, la capacità di distinguere tra diversi livelli logici diventa essenziale per comprendere e gestire sistemi complessi. Bateson, 1955, 1956, 1967, 1970, 1972a, , trad. it, in Bateson, 2000.

¹⁷¹ "But above all I wish to designate the following as the most important among the numerous questions which can be asked with regard to the axioms: To prove that they are not contradictory, that is, that a finite number of logical steps based upon them can never lead to contradictory results. [...]. To show the significance of the problem from another point of view, I add the following observation: If contradictory attributes be assigned to a concept, I say, that mathematically the concept does not exist. So, for example, a real number whose square is -1 does not exist mathematically. [...]. It is rather a system of things whose mutual relations are governed by the axioms set up and for which all propositions, and only those, are true which can be derived from the axioms by a finite number of logical processes". Il testo si trova in Hilbert, 1902, che riporta il celebre discorso di Hilbert all'ICM a Parigi nel 1900.

¹⁷² "One common understanding of formalism in the philosophy of mathematics takes it as holding that mathematics is not a body of propositions representing an abstract sector of reality but is much more akin to a game, bringing with it no more commitment to an ontology of objects or properties than playing ludo or chess are normally thought to have."

Ciò che Hilbert proponeva era una “formalizzazione e assiomatizzazione totale dell’aritmetica”, per la quale la coerenza era la condizione fondamentale, pena la “non-esistenza” nel corrispondente grado di impegno ontologico, dei concetti matematici.

In modo coerente a quanto fatto con la geometria euclidea l’anno precedente, in cui la chiave era metterne in luce la natura astratta, Hilbert voleva “dimostrare con la pura logica che i teoremi seguivano dagli assiomi, senza l’influenza corruttrice di quello che possiamo «vedere» guardando una figura” [Davis, 2000, p. 117].

In un celebre aneddoto, riportato dal suo allievo Blumenthal, l’idea di Hilbert era quella per cui: «si deve poter dire in ogni momento – invece di punti, rette e piani – tavoli, sedie e boccali di birra» a patto che questi oggetti obbediscano agli assiomi [Davis, 2000, p. 117].

Il significato intuitivo degli oggetti matematici è irrilevante: ciò che conta sono solo le relazioni che soddisfano gli assiomi.

Questa idea hilbertiana - in una sorta di curioso, ma non casuale, parallelo con la mossa di Shannon, analizzata nel capitolo precedente, riguardo la teoria dell’*informazione* – consiste, nella geometria come nella problematizzazione della coerenza, nel “separare nettamente la sintassi dalla semantica, e nello scartare quest’ultima” [Berto, 2009, p.47]. La coerenza è garante della verità di queste manipolazioni tra entità astratte che si ottiene mediante un numero finito di passi logici.

Il concetto espresso nel porre il problema della coerenza rimarrà ben presente, anzi costituirà il cuore, del Programma di Hilbert¹⁷³, in cui “Egli propose infatti di risolvere la crisi dei fondamenti dimostrando metamatematicamente la coerenza dell’aritmetica formalizzata con metodi puramente finitari.” [Berto, 2009, p.50].

La *metamatemica* o *teoria della dimostrazione* era, secondo Hilbert, il tentativo di rifondazione della matematica basato sull’idea per cui in un sistema formale si potevano usare tutti i metodi matematici, ma le regole che li governavano (i metodi metamatematici) dovevano essere assolutamente sicure e prive di contraddizioni (per questo *finitari*). “Hilbert aveva posto la questione della coerenza dell’aritmetica al secondo posto nell’elenco presentato nel 1900 al

¹⁷³ Sul Programma di Hilbert di risoluzione metamatematica e formalistica della “crisi” dei fondamenti matematici si vedano Lolli 2011 e 2016; Zach 2023. Quanto al rapporto tra la proposta hilbertiana e i paradigmi concorrenti si veda Davis 2000, pp. 109-132, trad. it. Il principale di queste correnti di pensiero era l’«intuizionismo» di Weyl e Brouwer (1881-1966), teoria secondo cui la matematica, fondata sull’intuizione dei numeri naturali, è «una creazione della mente. La verità di un enunciato matematico può essere concepita solo attraverso una costruzione mentale che ne dimostri la validità» [“Intuitionism is based on the idea that mathematics is a creation of the mind. The truth of a mathematical statement can only be conceived via a mental construction that proves it to be true, and the communication between mathematicians only serves as a means to create the same mental process in different minds”, Lemhoff, 2019].

Congresso internazionale dei Matematici, ma cominciò ad affrontare sul serio il problema solo negli anni Venti” [Davis, 2000, pp. 127-128 trad. it.] lavorando a stretto contatto, tra gli altri, con William Ackermann (1896-1962).

In questo orizzonte “essendo così spogliate di ogni significato, le deduzioni di enunciati (che sono un certo tipo di formule, quelle che non contengono variabili non vincolate da quantificatori) all'interno del sistema formale diventano una procedura di tipo meccanico, ossia puramente combinatoria” [Berto, 2009, p.48]. La matematica, formalisticamente, anzi *metamaticamente*, diviene così paragonabile a un gioco. Una manipolazione formale, e coerente, di simboli. Le regole, i passaggi “leciti”, di una dimostrazione sono da considerare simili alle mosse possibili consentite in una partita di scacchi. “In questa analogia, «giocare a scacchi corrisponde a «fare della matematica»; ma gli enunciati fatti *intorno* al gioco degli scacchi, cioè quelli che entrano *nel merito* del gioco [...] corrispondono agli enunciati fatti *intorno* ai limiti di competenza della matematica” [Hodges, 1983, p.126 trad. it.] Il programma di Hilbert intendeva occuparsi di enunciati di questo genere.

Hilbert, nel 1928, proprio insieme a William Ackermann, diede alle stampe *Grundzüge der theoretischen Logik* [Fondamenti della logica teorica], basato sui corsi di logica che teneva dal 1917. In questo testo Hilbert afferma che “i sistemi logici potevano essere visti anche dall'esterno divenne possibile dare loro quella forma precisa in cui ora venivano enunciati” [Davis 2000, p.131 trad. it.]. La logica di prim'ordine – avviata con l'*Ideografia* di Frege [Frege, 1879] e proseguita con i *Principia Mathematica* di Russell e Whitehead – sollevava già problemi noti, ma proprio questa nuova prospettiva rese possibile dar loro concretezza e una formulazione precisa, a partire dalla questione della coerenza.

Nel suo nuovo programma, matematica e logica sarebbero state sviluppate insieme come linguaggio simbolico puramente formale, che poteva essere visto sia «dall'interno» sia «dall'esterno». Visto dall'interno non era che matematica, con ogni passo deduttivo, per quanto piccolo, totalmente esplicitato; visto dall'esterno era invece solo un insieme di formule e manipolazioni di simboli che potevano essere eseguite prescindendo da ogni significato. Hilbert intendeva dimostrare che in un tale linguaggio non si poteva derivare una coppia di formule contraddittorie, e quindi, in maniera esplicita, che non si poteva dimostrare una formula come $1=0$ e $0\neq 0$ [Davis 2000, p.128 trad. it.].

Nel testo del '28 scritto con Ackermann, prendendo le mosse da questo sguardo “dall'esterno” sulla matematica, Hilbert diede forma precisa a tre problemi che nascevano dal

corpo della logica di prim'ordine: 1) il *Problema della Coerenza* [*Widerspruchsfreiheit*] - come visto, nucleo tematico forte del pensiero hilbertiano - volto a dimostrare che non è possibile derivare in un sistema formale una proposizione e il suo contrario; 2) il *Problema della Completezza* [*Vollständigkeit*], in cui si sostiene che tutte le proposizioni valide in un sistema formale possono essere derivate dagli assiomi e dalle regole di inferenza; 3) il *Problema della Decisione* [*Entscheidungsproblem*], nel quale si mostra come, grazie a un numero limitato di passi e a un metodo effettivo, sia possibile determinare la verità o la falsità di una proposizione all'interno di un sistema formale [Hilbert, Ackermann, 1928, pp. 31-35, pp.94-97].

Gli stessi problemi – insieme alla critica del metodo intuizionista, alla difesa della formalizzazione e a riflessioni sull'impiego di oggetti ideali [*ideale Aussagen*] per estendere il pensiero matematico oltre i metodi finitari [Hilbert, 1928] – furono presentati alla comunità matematica mondiale riunitasi dal 3 all'11 settembre a Bologna per l'ottavo Congresso Internazionale dei Matematici. Un'edizione dell'ICM segnata dalle interferenze politiche e dalla frammentazione delle realtà scientifiche, conseguenze dirette della Prima Guerra Mondiale e, in particolare, del Trattato di Versailles.

Coerenza, completezza e decisione: tre problemi dai quali, per Hilbert, dipendeva il futuro stesso della scienza matematica. Essi non solo riflettevano le sue convinzioni matematiche, ma incarnavano lo spirito della sua impresa scientifica, sintetizzato nelle parole pronunciate a Parigi nel 1900: “Ecco il problema, trova la soluzione; la puoi trovare mediante il puro pensiero, perché in matematica non c'è l'*ignorabimus*” [Hilbert, 1902, trad. it. in Lolli, 2016, p. 114]. A quei problemi, tuttavia, non era stata ancora data risposta.

Il sorgere di questi problemi si radicava in uno “spirito meccanico”. L'assiomatizzazione, la riduzione degli enunciati alla sola dimensione sintattica e l'adozione di un punto di vista *esterno* delineano un'idea della matematica come mera manipolazione di simboli secondo regole. “Essendo così spogliate di ogni significato, le deduzioni di enunciati (che sono un certo tipo di formule, quelle che non contengono variabili non vincolate da quantificatori) all'interno del sistema formale diventano una procedura di tipo meccanico, ossia puramente combinatoria” [Berto, 2009, p.48].

È questa idea – e le tappe che condussero al collasso del Programma di Hilbert, esemplificato dai tre problemi in questione – a connettere idealmente la Parigi infuocata dell'estate che apre il Ventesimo secolo con una cena di metà anni '30 alla *High Table*, la tavola riservata ai *fellow* e ai don del *King's College* di Cambridge. Lì sedeva un ventiduenne che aveva appena ottenuto la *fellowship* per i suoi lavori sulla funzione gaussiana e sulla riscoperta del

teorema centrale del limite. Si trattava di Alan Turing [1912-1954], che stava conversando con un altro fellow, Richard Braithwaite¹⁷⁴ – filosofo della scienza e della religione – il quale era riuscito a strappargli qualche informazione sui temi di ricerca, nonostante la sua abituale reticenza. Turing - ispirato dalle lezioni di un altro personaggio che rincontreremo nel nostro racconto, Max Newman - stava allora lavorando alla risoluzione del terzo problema posto da Hilbert nel 1928: l'*Entscheidungsproblem*.

Il lavoro di Turing sulla risoluzione del Problema della Decisione avrebbe portato il matematico inglese alla stesura dell'articolo *On Computable Numbers, with an Application to the Entscheidungsproblem* [Turing, 1936] che ha i concetti chiave per lo sviluppo del calcolatore moderno e che ha posto le basi teoriche delle caratteristiche della nostra contemporaneità digitale segnata dal ruolo centrale del calcolatore.

È necessario, per la corretta comprensione dell'orizzonte concettuale in cui si staglia il lavoro di Turing, fare, purtroppo brevemente, riferimento alle rivoluzionarie conseguenze contenute in un oscuro articolo intitolato *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I* scritto da un logico austriaco allora venticinquenne, già, seppur con distacco frequentatore del Circolo di Vienna, Kurt Gödel [Edmonds, 2020, p.21].

Una figura centrale nella matematica, nella computazione e nell'impatto della scienza sulla sfera pubblica del XX secolo, come John von Neumann – che fu il primo a cogliere la portata rivoluzionaria della scoperta di Kurt Gödel¹⁷⁵ durante i lavori della Seconda Conferenza

¹⁷⁴ R. D. Braithwaite (1900-1990) è anche il presunto proprietario dell'attizzatoio che si narra Wittgenstein brandì sotto il naso di Karl Popper la sera del 25 ottobre 1946, durante una delle discussioni più famose della storia della filosofia, tenutasi al *Moral Science Club* di Cambridge. Il dibattito verteva sull'esistenza di problemi filosofici autentici e vide la presenza, oltre a Wittgenstein e Popper, anche di Bertrand Russell. Per l'episodio, emblematico di un tratto decisivo della storia del pensiero filosofico e scientifico del Novecento, si veda Edmonds, Eidinow, 2001.

¹⁷⁵ von Neumann si spese enormemente per portare Gödel in salvo negli Stati Uniti e garantirgli una posizione all'*Institute for Advanced Study* (IAS) dopo l'Anschluss dell'Austria nel 1938. In una lettera inviata ai vertici della diplomazia, tramite i buoni uffici dei vertici dello IAS e della Rockefeller Foundation, scrisse: "Gödel è assolutamente insostituibile: è l'unico matematico vivente a proposito del quale oserei fare tale affermazione» «Salvarlo dal tracollo dell'Europa è uno dei contributi più grandi che chiunque possa dare" (Dyson, 2012, p. 111, trad. it). Queste dichiarazioni furono parte di un appello che, tramite i vertici dello IAS e della Rockefeller Foundation, von Neumann fece circolare nelle alte sfere diplomatiche per assicurarsi che Gödel ottenesse il visto per gli Stati Uniti e una posizione accademica a Princeton. Gödel arrivò a Princeton nel 1940, trovando rifugio accademico e umano nell'amicizia con Albert Einstein - che si può vedere anche in una breve scena del film di Christopher Nolan del 2023, *Oppenheimer* - che lo accompagnava spesso nelle sue passeggiate. Tuttavia, la sua vicenda biografica assunse toni sempre più tragici: afflitto da gravi problemi mentali, sviluppò un'ossessione patologica per il cibo e la paura di essere avvelenato. Nel 1978 morì per inedia, ridotto a un peso di circa 30 kg, incapace di nutrirsi se non da sua moglie, che si trovava ricoverata in ospedale. Per la vicenda biografica di Gödel si veda Dawson, 1997. Sul periodo di Gödel a Princeton presso l'*Institute for Advanced Studies* si veda Dyson, 2012, pp. 103-124, trad. it. In Gleick, 2011 si mostrano i risultati del logico austriaco in relazione alla storia dello sviluppo della teoria dell'informazione. L'orizzonte intellettuale dei decisivi contributi di Gödel alla logica e alla matematica, il loro peso filosofico e la relazione tra le scoperte di logica matematica e le implicazioni filosofiche sono trattati in

sull'Epistemologia delle Scienze Esatte, tenutasi nel settembre 1930 a Königsberg [Davis, 2000, pp.154-155 trad. it.; Dyson, 2012, p.109 trad. it.] – avrebbe poi definito così i risultati del logico austriaco:

La conquista di Gödel nel campo della logica moderna è unica e monumentale – anzi, è più di un monumento, è una pietra miliare che rimarrà visibile a lungo nello spazio e nel tempo [...] Il risultato è straordinario nella sua quasi paradossale 'negazione di sé': con gli strumenti della matematica non sarà mai possibile ottenere la certezza che la matematica non contiene contraddizioni [...]. La logica non sarà mai più la stessa [Dyson, 2012, p.109 trad. it.].

Il giudizio di von Neumann coglie l'essenza del contributo di Gödel: la possibilità di costruire “entro un sistema formale logico-matematico enunciati di tipo autoreferenziale senza «riferimenti empirici» di alcun genere: enunciati autoreferenziali il cui contenuto è tanto poco empirico e contestuale quanto quello dell'enunciato” [Berto, 2009, p.12]. In altri termini, il logico austriaco mostrò come, metamaticamente, mediante l'aritmetica stessa, fosse possibile esprimere proprietà del sistema formale – identificato con i *Principia Mathematica* di Russell e Whitehead, indicati con PM – e, attraverso di esse, rivelarne incompletezza e incoerenza, infliggendo dall'interno un colpo mortale alle pretese del Programma di Hilbert.

La chiave di questo risultato è stata un'operazione di codifica, intendendo per quest'ultima una funzione $\varphi: D \rightarrow N$ che associa un numero naturale a ciascun elemento di un insieme D , permettendo di rappresentare simboli ed espressioni linguistiche o matematiche mediante numeri naturali [Frixione, Palladino, 2011, p.39].

Gödel operò uno speciale tipo di codifica, la gödelizzazione:

Una funzione g che abbina a ogni simbolo, formula e sequenza di formule del sistema formale in gioco un singolo numero naturale, in modo univoco ed effettivo. Data un'espressione (simbolo, formula, sequenza di formule) del sistema, la funzione le assegna il numero $g(e)$, che è detto il suo numero di Gödel, o brevemente il suo gödeliano [Berto, 2009, p.77].

Lolli, 1992, 2019; Berto, 2009. Una disamina dei risultati gödeliani accessibile ai non specialisti è in Nagel, Newman, 1958. Si veda anche H. Wang, 1974, testo che gode del punto di vista privilegiato della vicinanza umana dell'autore con Gödel.

Mediante la gödelizzazione, tutti i simboli di PM e le relative regole sintattiche venivano rappresentati numericamente, così che qualsiasi espressione matematica potesse essere tradotta in una sequenza di numeri. Se, metamaticamente, un simbolo equivale a un altro, allora si può utilizzare qualunque segno – lettere o notazioni logiche – ma Gödel scelse di impiegare i numeri. La gödelizzazione è infatti un procedimento algoritmico applicabile in modo meccanico, senza alcuna comprensione del contenuto semantico. Fu così inflitto un colpo al cuore del programma formalista proprio attraverso la più acuminata delle sue armi: la rinuncia ad ogni dimensione che non fosse puramente sintattica. E fu dal cuore dei numeri che egli mostrò come, nell'aritmetica e “in mezzo a PM (e in qualsiasi sistema coerente di logica) si debbono nascondere dei mostri di un tipo fin qui inaudito” [Gleick, 2011, p.170 trad. it.].

Dall'applicazione di questo procedimento, Gödel dimostrò che vi è più verità di quella che un sistema formale riesce a formalizzare completamente. Vi erano enunciati veri espressi nel sistema formale che eccedevano il sistema stesso e le sue regole. Quindi questo era *incompleto*. "Gödel mostrò in che modo sia possibile costruire una formula aritmetica G che rappresenti la proposizione metamatematica: 'la formula G non è dimostrabile'. Questa formula, allora, evidentemente afferma la *sua* non dimostrabilità" [Nagel, Newman, 1958, p.93 trad. it.]

Dall'interno di PM, cioè, emerge un enunciato che Russell, attraverso la teoria dei tipi, aveva cercato di evitare a tutti i costi: l'enunciato autoreferenziale. Quel tipo di proposizione che, nelle diverse maschere assunte – dal mentitore cretese al barbiere del villaggio – ha tormentato il pensiero filosofico sin dall'antichità, ora riemergeva dalle viscere del sistema nato per esorcizzarlo:

$$R(q) \rightarrow q \text{ "io non sono dimostrabile"}$$

Questa formula, G , costruita da Gödel “di fatto, afferma: «La proposizione con il numero di Gödel g non si può dimostrare» [...] Nel sistema stabilito, G non si può dimostrare, dunque è vera”. [Dyson, 2012 p.110 trad. it.]

Gödel espose come G è dimostrabile se, e solo se, la sua negazione formale $\neg G$ è dimostrabile. Ma se una formula e la sua negazione sono entrambe formalmente dimostrabili, allora il sistema è contraddittorio. Dunque, se il sistema è coerente, né G né $\neg G$ risultano essere formalmente deducibili dagli assiomi dell'aritmetica. [Nagel, Newman, 1958, pp.93 trad.it.].

Come Gödel stesso ha premura di sottolineare, non si tratta di un caso di circolarità tra metalinguaggio e linguaggio-oggetto, che dà origine ai paradossi nel linguaggio ordinario

secondo la teoria della verità di Tarski [Tarski, 1944; Berto, 2009, pp.181-185, p.189]. Ciò è reso possibile proprio dal fatto che, grazie alla gödelizzazione, tutto viene tradotto in termini numerici.

A dispetto delle apparenze, una tale proposizione non contiene circoli viziosi, perché inizialmente essa afferma soltanto che una certa formula ben definita [...] non è dimostrabile. Solo in seguito (e per così dire, per caso) vien fuori che questa formula è precisamente quella mediante la quale era stata espressa la proposizione stessa. [Gödel, 1931, p.175 n.15 trad. mia]¹⁷⁶

A partire da queste basi di Gödel può formulare il Primo Teorema di Incompletezza in quale, alla lettera, afferma:

Per ogni classe ricorsiva ω -coerente X di formule, esiste un segno di classe ricorsivo r , tale che né $\forall v \text{ Gen } r$ né $\neg(\forall v \text{ Gen } r)$ appartengono a $\text{Flg}(X)$ [Gödel, 1931, Satz VI, p.187 trad. mia]¹⁷⁷.

In altre parole, ogni sistema formale coerente e sufficientemente potente da esprimere l'aritmetica contiene proposizioni vere che non possono essere dimostrate al suo interno. Si tratta di un'*incompletezza* strutturale del sistema stesso.

Dato che G è vera e insieme formalmente indecidibile, gli assiomi dell'aritmetica *non sono completi*. In altre parole, non possiamo dedurre tutte le verità aritmetiche dagli assiomi. Inoltre, Gödel dimostrò che l'aritmetica è *essenzialmente* incompleta: anche se si supponessero altri assiomi aggiuntivi, tali da permettere la formale deduzione della formula vera G dall'insieme più ampio, si potrebbe costruire un'altra formula vera ma formalmente indecidibile [Nagel, Newman, 1958, p.94 trad. it.].

Non solo esistono verità matematiche, come l'enunciato G , che non sono dimostrabili all'interno del sistema, ma questo, indipendentemente da quanti nuovi assiomi aggiungiamo, sarà sempre incompleto. È, infatti, sempre possibile costruire enunciati del tipo G .

Non pago di questo risultato già di per sé clamoroso, "Gödel, infine, descrisse la maniera in cui costruire una formula aritmetica A che rappresenti la proposizione metamatematica:

¹⁷⁶ "Ein solcher Satz hat entgegen dem Anschein nichts Zirkelhaftes an sich, denn er behauptet zunächst die Unbeweisbarkeit einer ganz bestimmten Formel (nämlich der q -ten in der lexikographischen Anordnung bei einer bestimmten Einsetzung), und erst nachträglich (gewissermaßen zufällig) stellt sich heraus, daß diese Formel gerade die ist, in der er selbst ausgedrückt wurde".

¹⁷⁷ "Zu jeder ω -widerspruchsfreien rekursiven Klasse κ von Formeln gibt es rekursive Klassenzeichen r , so daß weder $\forall v \text{ Gen } r$ noch $(\neg(\forall v \text{ Gen } r))$ zu $\text{Flg}(\kappa)$ gehört (wobei vv die freie Variable aus rr ist)".

'l'aritmetica è coerente'; e dimostrò che la formula ' $A \supset G$ ' [equivalente a $\neg A \Rightarrow G$] è formalmente dimostrabile. Finalmente, egli mostrò che la formula A non è dimostrabile". [Nagel, Newman, 1958, p.94 trad. it.]. Cioè, il calcolo aritmetico formale non è in grado di dimostrare la coerenza dell'aritmetica. L'edificio di Hilbert crolla così proprio sul suo pilastro portante: quell'esigenza di coerenza che aveva animato il suo Programma fin dalla conferenza ICM di Parigi del 1900.

Nessun sistema formale può essere allo stesso tempo completo (capace di dimostrare tutte le verità) e coerente (privo di contraddizioni). Gödel quindi ha gioco facile a enunciare il suo Secondo Teorema:

Sia X una qualsiasi classe ricorsiva e consistente di formule. Allora la formula che afferma che X è coerente non è dimostrabile in X . In particolare, la consistenza di P non è dimostrabile in P , a condizione che P sia coerente [Gödel, 1931, Satz XI, p.196 trad. mia]¹⁷⁸.

Vale a dire che nessun sistema formale potente al punto da contenere l'aritmetica può essere usato per dimostrare la propria coerenza. Così, le prime due domande del programma di Hilbert ricevono da Gödel una risposta negativa.

2. Algoritmi dell'Indecisione

Il terzo problema di Hilbert chiamava in causa la portata del programma del matematico di Göttinga. "Hilbert [...] cercava un algoritmo di un'ampiezza senza precedenti; in linea di principio l'algoritmo per l'*Entscheidungsproblem* avrebbe dovuto ridurre tutti i ragionamenti deduttivi umani a calcolo bruto" [Davis, 2000, p. 182, trad. it.].

Un *don* di Cambridge, il matematico Max Newman (1897-1984), nel 1928 aveva ascoltato Hilbert porre i suoi problemi al Congresso di Bologna. "Affascinato da queste novità, nella primavera del 1935 tenne un corso sui fondamenti della matematica che culminava col teorema di incompletezza di Gödel" [Davis, 2000, p. 183]. In questo corso l'argomento era "l'approccio sistematico" [Copeland, 2012, p. 16, trad. it.], ovvero un metodo che procedeva passo dopo passo secondo regole precise, riflettendo l'obiettivo della formalizzazione e dell'assiomatizzazione della matematica tentate da Hilbert. "Newman spiegò al pubblico che la

¹⁷⁸ "Sei κ eine beliebige rekursive widerspruchsfreie Klasse von Formeln, dann gilt: Die Satzformel, welche besagt, daß κ widerspruchsfrei ist, ist nicht κ -beweisbar; insbesondere ist die Widerspruchsfreiheit von P in P unbeweisbar, vorausgesetzt, daß P widerspruchsfrei ist (im entgegengesetzten Fall ist natürlich jede Aussage beweisbar)".

caratteristica base di tutti questi processi sistematici è che una macchina è in grado di eseguirli”¹⁷⁹ [Copeland, 2012, p. 16, trad. it.].

Queste parole accesero la fantasia di un giovane matematico di Cambridge, lo stesso che poche pagine addietro abbiamo incontrato seduto alla *High Table* dell’università britannica: Alan Turing. “Ciò che Turing sentì durante le lezioni di Newman lo affascinò a tal punto da dominare la sua attività lavorativa per mesi” [Copeland, 2012, p. 17 trad. it.].

Quello a cui stava lavorando Turing era un attacco al problema della decisione (*Entscheidungsproblem*) a partire da una prospettiva del tutto nuova, che sembrava un’estremizzazione del programma hilbertiano. I suoi sforzi culminarono nell’aprile del 1936 in una bozza del fondamentale saggio *On Computable Numbers, with an Application to the Entscheidungsproblem* [Turing, 1936].

I risultati del lavoro di Turing in questo saggio sono fondamentali per almeno tre motivi:

1. La sua trattazione dell’*Entscheidungsproblem* ha contribuito alla definizione rigorosa di algoritmo.
2. La soluzione da egli escogitata ha gettato le fondamenta teoriche dell’informatica moderna e del moderno computer.
3. Ha creato la piattaforma per un nuovo paradigma culminato nell’Intelligenza Artificiale Classica [GOFAI], nel programma di ricerca delle scienze cognitive e nel funzionalismo in filosofia della mente.

Quando Turing consegnò a Newman il suo lavoro, quest’ultimo “dovette avere uno shock” [Copeland, 2012, p. 18 trad. it.]. Turing, in quelle pagine, aveva inventato una macchina universale.

Procediamo per gradi. Secondo Turing, “ogni calcolo può essere inteso come un processo con le seguenti caratteristiche:

- Viene eseguito scrivendo dei simboli nelle caselle di un nastro di carta.
- A ogni passo la persona che esegue il calcolo fa attenzione al simbolo scritto in una sola di queste caselle.
- L’azione successiva dipenderà da questo simbolo e dallo stato mentale della persona.

¹⁷⁹ Max Newman fu protagonista di un risultato che dava corpo alle idee di Turing, che lui stesso aveva contribuito a ispirare. Newman ebbe un ruolo centrale nella costruzione del *Manchester Baby* e delle altre macchine di Manchester (che incontreremo più avanti). il primo computer elettronico a programma memorizzato, sebbene il primato sia talvolta conteso con altri progetti. Completato il 21 giugno 1948, incorporava memoria a tubo catodico e un’architettura che permetteva l’esecuzione dinamica di istruzioni, realizzando per la prima volta l’idea di un calcolo completamente automatizzato.

- Tale azione consisterà nello scrivere un simbolo nella casella osservata ed eventualmente nello spostare l'attenzione sulla casella immediatamente a destra o a sinistra.

È facile vedere come una persona possa essere sostituita da una macchina" [Davis, 2000, p. 187, trad. it.].

A partire da questa considerazione, Turing, per risolvere il problema della calcolabilità ed elaborare una procedura effettiva, concepì una serie di macchine astratte. Dopo aver definito le macchine automatiche come quelle "totalmente determinate dalla configurazione" e le macchine calcolatrici come quelle che stampano solo due tipi di simboli (0 e 1), propose di utilizzare un determinato tipo di macchine [Turing, 1936, p.232]¹⁸⁰, *astratte*, che definiremo le *Macchine di Turing* (MT).

Queste macchine sono astratte perché non tengono conto, sebbene vedremo come in esse vi sia in potenza una tendenza alla loro implementazione reale, dei vincoli pratici fondamentali per la progettazione di un calcolatore reale, come le dimensioni della memoria o i tempi necessari per l'elaborazione. Inoltre, la loro definizione è indipendente dalla natura della loro realizzazione fisica: non conta il tipo di hardware utilizzato, ma solo le relazioni che sussistono tra le loro parti. Proprio questa caratteristica le renderà particolarmente rilevanti per il funzionalismo filosofico e le scienze cognitive, poiché suggerisce che il concetto di calcolo non dipende dal supporto materiale, ma dalla struttura formale del processo computazionale

Turing sapeva che un algoritmo è tipicamente definito da un elenco di regole che una persona può seguire in modo meccanico e preciso, come le ricette nei libri di cucina. Tuttavia, spostò la sua attenzione dalle regole a ciò che la persona *effettivamente* faceva calcolando. Eliminando progressivamente i dettagli inessenziali, riuscì a dimostrare che un tale processo poteva essere formalizzato.

A partire da questa presa di posizione metodologica, Turing si propose di stabilire le condizioni "rigorose" entro cui è possibile calcolare un numero: "I numeri calcolabili sono quelli le cui espressioni decimali sono calcolabili con mezzi finiti" [Turing, 1936, p.231 trad. mia]¹⁸¹, motivazione che egli collegò al fatto che "la memoria umana è necessariamente limitata"

¹⁸⁰ "If at each stage the motion of a machine [...] is *completely* determined by the configuration, we shall call the machine an "automatic machine" (or *a-machine*) ; If an *a-machine* prints two kinds of symbols, of which the first kind (called figures) consists entirely of 0 and 1 (the others being called symbols of the second kind), then the machine will be called a computing machine".

¹⁸¹ "We have said that the computable numbers are those whose decimals are calculable by finite means".

[Turing, 1936, p.231 trad. mia]¹⁸². L'essere umano impegnato nell'effettiva impresa di calcolo di un numero reale viene messo a confronto con una macchina calcolatrice automatica del tipo sopra definito. Macchina che è dotata, determinata e guidata da una serie di condizioni iniziali.

Ma quali sono le caratteristiche di questa macchina?

La macchina è fornita di un 'nastro' (l'analogo della carta) che scorre attraverso di essa, diviso in sezioni (chiamate 'quadrati'), ciascuna capace di portare un 'simbolo'. In ogni momento c'è un solo quadrato, diciamo l'-esimo, che porta il simbolo che è 'nella macchina'. Possiamo chiamare questo quadrato il 'quadrato scandito'. Il simbolo sul quadrato scandito può essere chiamato il 'simbolo scandito' [Turing, 1936, 231 trad. mia]¹⁸³.

Un essere umano, nel calcolo, opera solitamente su un supporto di carta bidimensionale. Tuttavia, questa caratteristica non è essenziale: Turing osserva che, senza perdere di generalità, possiamo supporre un nastro monodimensionale, il che non comporta alcuna perdita di potenza espressiva. Possiamo quindi considerare "un nastro monodimensionale di grandezza potenzialmente infinita" [Frixione, Palladino, 2011, p.51], suddiviso, come i quaderni di aritmetica delle scuole elementari, in quadrati, ciascuno capace di contenere un simbolo. In ogni determinato passaggio del calcolo si prende in considerazione solo uno di questi quadrati: il *quadrato scandito*, all'interno del quale si trova un solo simbolo, il *simbolo scandito*.

Per quanto riguarda i simboli utilizzati nel calcolo, "ogni macchina potrà disporre soltanto di un insieme finito di essi, che costituirà l'alfabeto di quella macchina (e che chiameremo Σ)" [Ibidem]. Il fatto che l'alfabeto sia finito non rappresenta una limitazione. Sia dunque $\Sigma = \{s_1, s_2, \dots, s_n\}$ l'alfabeto di una generica *Macchina di Turing*. Ogni cella del nastro potrà contenere uno di tali simboli oppure, in alternativa, restare vuota (indicheremo la cella vuota con s_0) [Frixione, Palladino, 2011, p.52].

"Possiamo paragonare un uomo nel processo di calcolo di un numero reale a una macchina che è capace solo di un numero finito di condizioni, che chiameremo 'configurazioni'" [Turing, 1936, p.231 trad. mia]¹⁸⁴. Come si è accennato, il punto di partenza di Turing è

¹⁸² "For the present I shall only say that the justification lies in the fact that the human memory is necessarily limited".

¹⁸³ "The machine is supplied with a "tape" (the analogue of paper) running through it, and divided into sections (called "squares") each capable of bearing a "symbol". At any moment there is just one square, say the r -th, bearing the symbol $s_{(r)}$, which is "in the machine". We may call this square the "scanned square". The symbol on the scanned square may be called the "scanned symbol".

¹⁸⁴ "We may compare a man in the process of computing a real number a to a machine which is only capable of a finite number of conditions $q_1, q_2, \dots, q_n$ which will be called " m -configurations".

l'osservazione delle operazioni effettive che un essere umano compie nel calcolo. Questo implica che vi sia sempre una porzione limitata di nastro osservabile in un dato istante. Turing assume, senza perdita di generalità, che la macchina possa esaminare soltanto una cella alla volta e "osservare" a ogni passo al massimo un singolo simbolo.

A tal scopo, la macchina sarà dotata di una testina di lettura/scrittura, chiamata Δ , che può operare muovendosi verso destra e verso sinistra

Chi sta eseguendo un calcolo ha poi la possibilità di scrivere nuovi simboli, di cancellare quelli già scritti o di sostituirli con altri. La testina eseguirà anche tali compiti di cancellazione e di scrittura. Anche in questo caso, però, essa potrà agire soltanto sulla cella 'osservata' e, per accedere ad altre celle, dovrà prima spostarsi lungo il nastro. Poiché ogni cella può contenere un solo simbolo, quando in una cella si scrive un nuovo simbolo, il simbolo eventualmente già presente verrà cancellato [Frixione, Palladino, 2011, p.52].

Poiché il punto di partenza è l'esecuzione umana del calcolo, il calcolatore umano deve tenere conto delle operazioni che ha compiuto e dei simboli osservati e memorizzati. Nell'atto del calcolo è dunque centrale lo "stato mentale" [Turing, 1936, p.250 trad. mia]¹⁸⁵ di chi calcola.

Turing suppose che una macchina potesse assumere un certo numero di stati interni, corrispondenti agli stati mentali dell'essere umano, e che tali stati siano in numero finito. "Siano allora $q_1 \dots q_{(r_n)}$ gli stati che una generica MT può assumere" [Frixione, Palladino, 2011, p.52].

La *Macchina di Turing* [fig.10], analogamente al calcolatore umano, è definita dal suo stato in un determinato momento del calcolo, ovvero dalla sua "configurazione": "Il comportamento possibile della macchina in ogni momento è determinato dalla configurazione e dal simbolo scandito" [Turing, 1936, p.231 trad. mia]¹⁸⁶.

In precedenza, abbiamo notato come Turing identifichi nella configurazione le condizioni che determinano l'operatività della macchina, ma quali sono queste condizioni? Esse sono individuate dalla coppia (q_n, s_n) , dove q_n rappresenta lo stato attuale della macchina tra quelli possibili e s_n è il simbolo presente nel quadrato attualmente esaminato.

Questa coppia sarà chiamata 'configurazione': quindi la configurazione determina il comportamento possibile della macchina. In alcune delle configurazioni in cui il quadrato scandito è

¹⁸⁵ "state of mind".

¹⁸⁶ "The possible behaviour of the machine at any moment is determined by the m –configuration q_n and the scanned symbol $s_{(r)}$ ".

vuoto (cioè non porta alcun simbolo), la macchina scrive un nuovo simbolo sul quadrato scandito: in altre configurazioni cancella il simbolo scandito. La macchina può anche cambiare il quadrato che viene scandito, ma solo spostandolo di un posto a destra o a sinistra. Oltre a una di queste operazioni, la configurazione può essere cambiata [Turing, 1936, p.231 trad. mia]¹⁸⁷.

È possibile scomporre l'operazione descritta in operazioni più semplici, in "operazioni atomiche" [Frixione, Palladino, 2011, p.54]: la sostituzione del simbolo osservato con un altro simbolo, lo spostamento della testina su una delle celle attigue del nastro e l'eventuale cambiamento dello stato interno della macchina [Frixione, 1992, p.10].

Rappresentazione di una Macchina di Turing (MT)

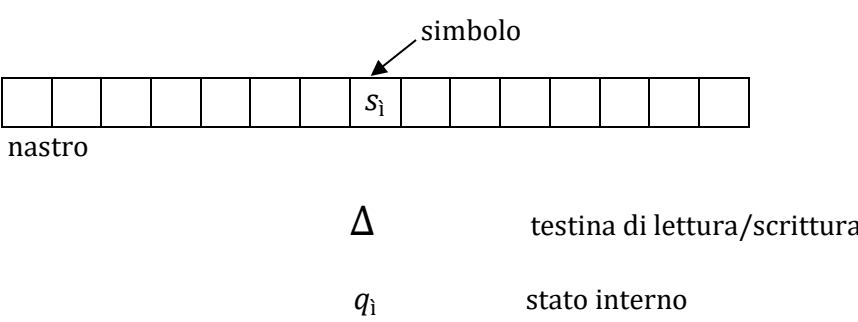


Fig.10 La Macchina di Turing

Indicate con le iniziali italiane D , C , S rispettivamente lo spostamento a destra, il rimanere al centro e lo spostarsi a sinistra, è possibile rappresentare un'operazione atomica attraverso una terna. Ad esempio, la terna $(s_i, D, q_{(r)})$ indica l'operazione di scrivere il simbolo s_i nella cella osservata, spostarsi a destra e assumere lo stato interno q_j .

Ogni Macchina di Turing (MT) deve poter essere programmata per eseguire un calcolo, intendendo per quest'ultimo "una serie di istruzioni che permettono di eseguire il compito per

¹⁸⁷ "This pair $q_n, s_{(r)}$ will be called the " configuration": thus the configuration determines the possible behaviour of the machine. In some of the configurations in which the scanned square is blank (i.e. bears no symbol) the machine writes down a new symbol on the scanned square: in other configurations it erases the scanned symbol. The machine may also change the square which is being scanned, but only by shifting it one place to right or left. In addition to any of these operationsm-configuration may be changed".

il quale è stata progettata" [Frixione, Palladino, 2011, p.54]. Ciò che nel calcolatore umano dipende dal ricordo delle operazioni eseguite e dai simboli osservati (ed è da questi vincolato), nella macchina ogni passo è deterministico e dipende esclusivamente dallo stato in cui si trova e dal simbolo letto, secondo una regola predefinita. Inoltre, ciò che la macchina *sa* è il simbolo nel quadrato scandito, "l'unico", aggiunge Turing, "di cui la macchina è, per così dire, direttamente consapevole" [Turing, 1936, p.231 trad. mia]¹⁸⁸.

Posta la scomponibilità "atomica" di ogni operazione, di conseguenza, ogni istruzione di una MT avrà la forma seguente:

Configurazione → Operazione atomica [Frixione, Palladino, 2011, p.54].

Quindi, ciò che un'istruzione della MT deve fare è specificare un'operazione atomica in riferimento a una specifica configurazione.

Nell'operazione descritta sopra, la macchina di Turing si trova nello stato q_i e legge il simbolo s_i , allora scrive s_i' , sulla cella attuale, si sposta a destra (D) e passa al nuovo stato q_i' .

$$q_i, s_i \rightarrow s_i', D, q_i'$$

Di questa *quintupla*, q_i, s_i rappresentano la configurazione, e cioè, Lo stato attuale della macchina e il simbolo attualmente letto sulla cella, mentre s_i', D, q_i' costituiscono l'operazione atomica, ossia l'azione che la macchina esegue. Si trova nello stato q_i e legge il simbolo s_i , allora scrive s_i' , sulla cella attuale, si sposta a destra D e passa al nuovo stato q_i

L'insieme delle quintuple costituisce le istruzioni per far eseguire alla macchina i calcoli per cui è stata progettata. Questo insieme è detto la "tavola della MT" [Frixione, Palladino, 2011, p.55]¹⁸⁹.

"Una macchina di Turing è un insieme finito (ma non vuoto) di quintuple che non contiene nessuna coppia di quintuple che coincidano nei loro primi due simboli" [Cordeschi, Tamburrini, 2001, p.20].

¹⁸⁸ "The "scanned symbol" is the only one of which the machine is, so to speak, "directly aware"".

¹⁸⁹ Perché un insieme di quintuple possa costituire una tavola valida per una Macchina di Turing, deve rispettare la condizione di determinismo: ogni configurazione della macchina, cioè una coppia composta da uno stato interno e un simbolo letto sul nastro, deve avere al massimo un'istruzione associata. Questo significa che, in ogni passo del calcolo, la macchina non può avere più opzioni di comportamento: dato un certo stato e un certo simbolo letto, l'azione da eseguire deve essere unica e ben definita. Un ulteriore aspetto riguarda la terminazione del calcolo. Perché la macchina possa fermarsi, devono esistere alcune configurazioni per le quali non è prevista alcuna istruzione. Se a ogni configurazione fosse sempre associata almeno una quintupla, la macchina continuerebbe ad operare all'infinito, senza mai raggiungere uno stato di arresto, Cfr. Frixione, Palladino, 2011, p.55.

Per Turing, l'insieme delle funzioni che possono essere calcolate da una MT corrisponde a tutte le funzioni calcolabili per mezzo di un algoritmo.

Se una macchina calcolatrice non scrive mai più di un numero finito di simboli del primo tipo, sarà chiamata circolare. Altrimenti, si dirà non circolare. Una macchina sarà circolare se raggiunge una configurazione dalla quale non è possibile alcuna mossa, o se continua a muoversi e possibilmente a stampare simboli del secondo tipo, ma non può stampare più simboli del primo tipo. Una sequenza è detta computabile se può essere calcolata da una macchina non circolare. Un numero è computabile se differisce per un numero intero dal numero calcolato da una macchina non circolare [Turing, 1936]¹⁹⁰.

Turing non si accontentò della “semplice” MT per sferrare il suo attacco all'*Entscheidungsproblem*. La macchina astratta - ispirata all'attività di calcolo di un essere umano - poteva dar conto del fatto che alcuni numeri o funzioni non fossero computabili. Tuttavia questo poteva non essere abbastanza. “Ma per passare da questo risultato alla conclusione che non esisteva nessun algoritmo, di nessun genere, capace di risolvere tale problema aveva dovuto utilizzare la propria analisi di quello che accade quando un essere umano esegue un computo” [Davis, 2000, p.202 trad. it.].

Ma quanto era affidabile, e generalizzabile, questa mossa? La risposta di Turing fu l'invenzione della Macchina Universale (MTU).

Abbiamo visto in precedenza l'operazione di codifica compiuta da Gödel per dimostrare il teorema di incompletezza, la gödelizzazione. Turing si mosse in modo simile, mostrando come fosse possibile trasformare le tavole che componevano le MT in una descrizione standard [*Standard Description*, S.D.], che a sua volta poteva essere tradotta in un numero di descrizione [*Description Number*, D.N.] [Turing, 1936, pp.240-241 trad. mia]¹⁹¹.

L'impiego di quello che a prima vista può sembrare un espediente tecnico aveva però un significato più profondo. Era la stessa idea che animava il risultato di Gödel: “e cioè la potente intuizione che essenzialmente non c'è distinzione fra «numeri» e operazioni sui numeri. Contemplando la matematica con occhi moderni, si tratta in entrambi i casi di simboli” [Hodges, 1985, p.141 trad. it.].

¹⁹⁰ “If a computing machine never writes down more than a finite number of symbols of the first kind, it will be called *circular*. Otherwise it is said to be *circle-free*. A machine will be circular if it reaches a configuration from which there is no possible move, or if it goes on moving, and possibly printing symbols of the second kind, but cannot print any more symbols of the first kind”.

¹⁹¹ “The integer represented by this numeral may be called a description number (D.N.) of the machine”.

Ogni *Macchina di Turing* può essere codificata e diventare un input per un'altra *Macchina di Turing*. La *Macchina Universale di Turing* (MTU) [fig.11] altro non è che una macchina "in grado di simulare il comportamento di ogni altra MT" [Frixione, 1992, p.12].

"Nonostante la sua estrema semplicità, la MUdT [MTU nella nostra notazione] è straordinariamente potente: vi sono ottime ragioni per credere che essa sia in grado di risolvere qualsiasi problema sia risolvibile da un algoritmo e viceversa" [Datteri, 2017, p.95]. Questo risultato prende il nome di *Tesi di Church-Turing* (o *Tesi di Church*)¹⁹², secondo cui le funzioni che possono essere calcolate da una MT sono tutte e sole le funzioni calcolabili per mezzo di un algoritmo [Frixione, 1992, p.12].

Questo rappresenta un passo decisivo nel passaggio da una definizione intuitiva di algoritmo, come procedimento passo dopo passo simile a una ricetta, a una definizione rigorosa: un algoritmo è una procedura composta da un insieme di regole che godono delle proprietà di *finitezza*, *determinismo* e *discretezza*.

Un algoritmo è una procedura finita, composta da un numero limitato di istruzioni che garantiscono l'ottenimento di un risultato. Deve essere deterministico, ossia, a parità di input, produce sempre lo stesso output seguendo passi ben definiti senza ambiguità. Inoltre, è discreto, cioè opera su dati finiti ed esegue istruzioni elementari in sequenza, senza processi continui o casuali. Queste caratteristiche lo rendono un metodo effettivo, applicabile sistematicamente senza bisogno di intuizione o creatività. Inoltre, deve essere sempre chiaro

¹⁹² Alonzo Church (14 giugno 1903 – 11 agosto 1995) fu un logico e matematico statunitense che, indipendentemente da Alan Turing, giunse a risultati fondamentali sulla computabilità e l'indecidibilità. Nel suo articolo "*An Undecidable Problem of Elementary Number Theory*", pubblicato nell'aprile 1936 sull'*American Journal of Mathematics* e leggermente antecedente a *On Computable Numbers* di Turing, dimostrò che non esiste un algoritmo generale per decidere la verità delle proposizioni aritmetiche, introducendo così il concetto di indecidibilità e basandosi sul λ -calcolo come modello di computabilità. Utilizzando il λ -calcolo, Church formalizzò il concetto di funzione computabile, quella per cui "una funzione degli interi positivi è effettivamente calcolabile solo se è λ -definibile (o, equivalentemente, ricorsiva)" [A function of positive integers is effectively calculable only if λ -definable (or, equivalently, recursive)]. [Copeland, 2023] mostrando che alcune funzioni non possono essere calcolate da alcun algoritmo, stabilendo così la non risolubilità dell'*Entscheidungsproblem* hilbertiano. Questo risultato è strettamente correlato a quello di Turing in *On Computable Numbers* (1936) [Frixione, Palladino, pp.111-115], poiché entrambi mostrarono limiti fondamentali della computazione meccanica: Church attraverso il λ -calcolo e Turing tramite le macchine di Turing, arrivando a conclusioni equivalenti sull'impossibilità di risolvere certi problemi in modo algoritmico. Ciò ha portato a identificare la loro concezione della computabilità con la Tesi di Church-Turing. Tuttavia, il risultato di Turing, anche al di là delle sue conseguenze pratiche (come l'idea di calcolatore moderno), è generalmente considerato superiore a quello di Church. Kurt Gödel vedeva lacune nell'approccio di Church, specialmente nel non aver dimostrato che non esiste alcun modo per costruire una "macchina delle decisioni", e ritenne il suo approccio tecnico "davvero molto insoddisfacente" [Copeland, 2012 pp. 27-28 trad. it.]. Al contrario, la soluzione di Turing era più chiara e convincente, collocando il problema nella giusta prospettiva (Wang, 1974, p. 25, cit. in Copeland, 2012, p. 18 trad. it.). Nonostante ciò, Church ebbe un ruolo centrale nella carriera di Turing, essendo il suo relatore di dottorato a Princeton. Sotto la sua supervisione, Turing scrisse la tesi "*Systems of Logic Based on Ordinals*" [Turing, 1939], completata e discussa nel maggio 1938, un lavoro che approfondì ulteriormente i limiti della computazione. Sulla Tesi Church-Turing si veda Copeland, 2023.

quando si è giunti al termine del procedimento, che può essere definito meccanico o eseguibile da una macchina [Datteri, 2017, p.96; Frixione, Palladino, 2011, pp. 11-23]¹⁹³.

Forte di questo risultato, Turing dimostrò che esistono “problemi matematici ben definiti che non possono essere risolti da una macchina calcolatrice finita” [Copeland, 2012, p. 20 trad. it]. Secondo la Tesi di Church-Turing, ciò significa che per questi problemi non esiste alcuna procedura algoritmica in grado di computarli.

Tra i problemi individuati da Turing vi è il problema della diagonalizzazione di Cantor, in cui applica il metodo della “*diagonal process*” [Turing, 1936, pp.246-249] per dimostrare che non tutte le sequenze numeriche sono computabili. Se si tentasse di elencare tutte le sequenze

¹⁹³ Questa definizione di algoritmo sembra paragonabile a quella data da R. K. Hill [Hill, 2015]: (1) “Un algoritmo è una struttura di controllo finita, astratta, efficace, composta, imperativamente data” (“An algorithm is a finite, abstract, effective, compound control structure, imperatively given”). Secondo questa definizione le caratteristiche fondamentali dell'algoritmo sono:

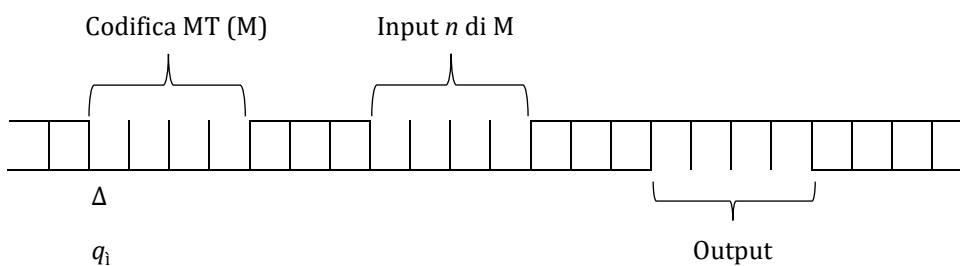
- Finitezza (*Finite*): Deve poter essere rappresentato in un tempo e spazio finiti.
- Astrattezza (*Abstract*): Non ha una collocazione spazio-temporale e può essere applicato a più istanze di un problema.
- Efficacia (*Effective*): Non richiede giudizi soggettivi, apprendimento o interpretazione; ogni passo è ben definito e garantisce un risultato.
- Controllo (*Control*): Definisce una transizione tra stati attraverso variabili e azioni, strutturandosi come una sequenza di operazioni.
- Struttura (*Structure*): Composto da più unità ordinate (parzialmente o linearmente), non riducibile a un singolo passaggio.
- Imperatività (*Imperative*): Fornisce istruzioni operative su come eseguire un compito, analogamente a una prescrizione o un comando.

Tuttavia, Hill non si limita a questa definizione, ma ne propone una seconda che introduce alcuni elementi ulteriori. La seconda definizione afferma: (2) “Un algoritmo è una struttura di controllo composta, finita, astratta, efficace, imperativamente data, che realizza uno scopo dato sotto condizioni date.” [“An algorithm is a finite, abstract, effective, compound control structure, imperatively given, accomplishing a given purpose under given provisions”]. Oltre alle proprietà già presenti nella prima definizione, questa versione allargata introduce due componenti fondamentali: scopo (*purpose*) e condizioni (*provisions*). L'algoritmo non è più solo una sequenza di operazioni astratte, ma è definito in relazione a un obiettivo specifico e a un insieme di vincoli che ne determinano l'applicazione. Questi elementi ancorano l'algoritmo a un contesto concreto, distinguendolo ulteriormente da una mera procedura formale. Ma vi è un ulteriore aspetto significativo che distingue (1) dalla definizione rigorosa di Turing. Un algoritmo non è semplicemente un insieme di regole descrittive, ma un sistema di istruzioni che devono essere eseguite in maniera operativa. Un algoritmo deve essere imperativamente dato, ovvero formulato in termini di azioni da compiere e non solo di stati o relazioni tra dati. Questa caratteristica lo distingue nettamente dalla macchina di Turing, che rimane un costrutto puramente formale e statico. “La macchina di Turing presuppone essenzialmente la presenza di un essere umano come parte del sistema: è l'essere umano a porre le domande, fornire le risorse materiali e interpretare le risposte.” (“The Turing Machine essentially presupposes a human as a part of a system-the human is the one who poses the questions, provides material resources and interprets the answers”) [Dodig-Crnkovic, 2011 in Hill, 2015]. La macchina di Turing, quindi, non è un'entità operativa autonoma, ma necessita di un agente esterno (umano o artificiale) che la interpreti e la metta in funzione. (1) con l'accento sulla natura fondamentale dell'imperatività come proprietà dell'algoritmo e l'allargamento di (2) a scopo e condizioni, solo amplia la definizione dell'algoritmo, ma supera anche la rigidità della macchina di Turing, distinguendo l'algoritmo come un oggetto computazionale attivo e orientato all'esecuzione di un compito. La definizione di Hill di algoritmo sembra più adatta a descrivere i modelli dinamici di *Machine Learning* (ML) e *Deep Learning* (DL), rispetto alla concezione classica basata sulla macchina di Turing, grazie all'enfasi nella definizione (2) su condizioni (*provisions*) e scopo (*purpose*). Nei sistemi ML/DL, inoltre, l'interazione con strutture di dati clusterizzati su larga scala sostituisce la necessità di un interprete umano sottolineata dalle MT, incorporando implicitamente il comportamento umano nelle fasi di addestramento e ottimizzazione del modello.

calcolabili da una Macchina di Turing e se ne costruisse una nuova modificando la cifra diagonale di ciascuna, si otterrebbe una sequenza che, per costruzione, non può essere nell'elenco. Questo mostra che, così come nel procedimento di Cantor esistono numeri reali non rappresentabili tra quelli razionali, esistono anche sequenze numeriche che non possono essere generate da alcun algoritmo, rendendole incomputabili.

Un altro esempio è il *problema del printing* [Turing, 1936]¹⁹⁴, per cui non esiste un metodo generale per stabilire se una Macchina di Turing stamperà mai un determinato simbolo, come "0", sul suo nastro. Poiché alcune macchine possono entrare in cicli infiniti senza mai produrre il simbolo atteso, il problema risulta indecidibile, rivelando un limite intrinseco della computazione e alla possibilità della formalizzazione integrale della matematica.

Fig.11 La macchina di Turing universale



3. Porridge freddo computazionale

Newman aveva illustrato al suo uditorio che i metodi hilbertiani, ovvero i processi sistematici in matematica, avevano una caratteristica fondamentale: la meccanizzazione del calcolo, l'idea che una "macchina" potesse eseguirli. Tuttavia, il risultato di Turing, aveva

¹⁹⁴ Il celebre problema dell'arresto (*halting problem*) non compare esplicitamente nell'articolo di Turing *On Computable Numbers* del 1936, bensì rappresenta una generalizzazione successiva del problema del *printing problem* posto da Turing. Quest'ultimo si riferisce alla possibilità per una macchina di stampare un simbolo su un nastro, mentre il problema dell'arresto concerne la determinazione della terminazione del calcolo. La formulazione esplicita dell'*halting problem* è dovuta a Martin Davis [Davis, 1958]. Il problema dell'arresto viene definito chiaramente come la questione di stabilire se una macchina di Turing, posta in un determinato stato iniziale, terminerà la computazione. Davis dimostra l'insolvibilità del problema mostrando che il predicato corrispondente non è computabile, utilizzando argomentazioni legate alla teoria della ricorsività e alla gödelizzazione. [Davis, 1958, p.70]. Si rimanda allo stesso testo per una disamina specialistica della teoria della computabilità.

“liquidato” l’*Entscheidungsproblem* [Hodges, 1985, p.536 trad. it; Davis, 2000, pp.198-202 trad. it.]. Paradossalmente, è stata proprio una macchina che fa crollare il sogno di Hilbert di un “unico, supremo processo sistematico per stabilire la verità o la falsità in matematica” [Copeland, 2012, p. 21 trad. it].

Hilbert vedeva nel lavoro di Frege e, soprattutto, in quello di Russell un passo decisivo nell’impresa sistematica di formalizzazione: “Nel compimento di questa grandiosa impresa russelliana dell’assiomatizzazione della logica si potrebbe vedere proprio il coronamento dell’opera dell’assiomatizzazione” [Lolli, 2011, p. 54].

Ironia della sorte, Poincaré aveva indirizzato una critica a Russell, estendibile all’intero programma di assiomatizzazione di Hilbert, per cui l’idea di fondare la matematica su un sistema di regole puramente formali, prive di qualsiasi riferimento intuitivo, portava inevitabilmente a una riduzione della disciplina a mero calcolo meccanico. Il matematico francese non perse occasione per ridicolizzare questa prospettiva:

Com'è facile capire, per dimostrare un teorema non sarà necessario e nemmeno utile sapere che cosa significa ... possiamo immaginare una macchina nella quale entrerebbero assiomi a un'estremità mentre dall'altra uscirebbero teoremi, un po' come quella leggendaria macchina di Chicago dove i maiali entrano vivi ed escono trasformati in salsicce e prosciutti [Poincaré in Davis, 2000 p. 122 trad. it.].

Eppure, proprio nel tentativo di smentire quel programma, si è giunti alla costruzione di una macchina. Il risultato è che, ancora oggi, essa alimenta la convinzione che ogni problema, purché formulato con sufficiente precisione, sia risolvibile. [Copeland, 2012, p. 24 trad. it.].

Nel 1937 Turing partì per gli Stati Uniti, alla volta di Princeton, per conseguire il suo PhD sotto la guida di Alonzo Church. La sua tesi di dottorato, accettata nel maggio del 1938, *Systems of Logic Based on Ordinals* [Turing, 1939], affronta il problema dell'incompletezza gödeliana e propone un metodo per ampliare i sistemi logici attraverso l'uso degli ordinali¹⁹⁵. “Il ben noto teorema di Gödel dimostra che ogni sistema logico è, in un certo senso, incompleto, ma allo stesso tempo indica dei mezzi attraverso i quali, a partire da un sistema logico L si può ottenere un sistema più completo L' Ripetendo il processo, si ottiene una sequenza $L, L_1 = L', L_2 = L_1'$ in cui ogni sistema è più completo del precedente” [Turing, 1939, p. 161]¹⁹⁶.

¹⁹⁵ Si vedano a riguardo, da un punto di vista matematico, Davis 2006; Feferman, 2006.

¹⁹⁶ “The well-known theorem of Gödel (Gödel [1], [2]) shows that every system of logic is in a certain sense incomplete, but at the same time it indicates means whereby from a system L of logic a more complete system L'

Il teorema di Gödel dimostra che ogni sistema formale sufficientemente potente è necessariamente incompleto: esistono sempre proposizioni vere ma non dimostrabili all'interno del sistema stesso. Un modo per sfuggire a questa limitazione consiste nell'aggiungere nuovi assiomi, in modo tale da permettere la dimostrazione di alcune di queste proposizioni. "Tuttavia, vista in questo modo, l'aritmetica assumeva inopinatamente l'aspetto di un'idra dalle molte teste" [Hodges, 1985, p.190 trad. it].

L'idea di Turing era quella per cui bisognava prendere in considerazione la possibilità di aggiungere una "infinità di assiomi" [Hodges, 1985, p.190 trad. it].

Si può allora costruire una logica L_ω in cui i teoremi dimostrabili sono la totalità dei teoremi dimostrabili con l'ausilio delle logiche $L, L_1, L_2 \dots$. Possiamo poi formare $L_{2\omega}$ che è correlato a L_ω , nello stesso modo in cui L_ω era correlato a L . Procedendo in questo modo, possiamo associare un sistema logico a qualsiasi ordinale costruttivo α . Ci si può chiedere se una sequenza di logiche di questo tipo sia completa, nel senso che a ogni problema A corrisponda un ordinale α tale che A sia risolvibile per mezzo della logica $L_{\omega\alpha}$ [Turing, 1939, pp.161-162]¹⁹⁷.

Turing voleva indagare se esistesse una lista di ordinali tale da permettere di sfuggire al teorema di Gödel. Tuttavia, qualunque fosse il numero finito di assiomi aggiunti, l'incompletezza continuava a manifestarsi: ogni volta che si ampliava un sistema per renderlo più completo, il teorema di Gödel tornava a colpirlo, generando sempre un nuovo enunciato vero ma indimostrabile.

Il matematico inglese denominò nella sua tesi "logica ordinale" [*ordinal logic*] [Turing, 1939, p. 189] l'insieme di queste aggiunte iterative di assiomi. Il problema - poiché nessuna logica può essere completa nel senso assiomatico-formalistico, non è possibile elencare tutti gli elementi di una lista infinita e persino un'eventuale regola di generazione ricadrebbe sotto la scure di Gödel - si presentava nei seguenti termini: "la regola per generare gli assiomi veniva data in termini di sostituzione di una certa espressione con una «formula ordinale»; il che era in sé stesso un «processo meccanico»" [Hodges, 1985, pp.191 trad. it].

may be obtained. By repeating the process we get a sequence $L, L_1 = L', L_2 = L_1' \dots$ each more complete than the preceding".

¹⁹⁷ "A logic L_ω may then be constructed in which the provable theorems are the totality of theorems provable with the help of the logics $L, L_1, L_2 \dots$. We may then form $L_{2\omega}$ related to L_ω , in the same way as L_ω was related to L . Ci si può chiedere se una sequenza di logiche di questo tipo sia completa, nel senso che a ogni problema A there corresponds an ordinal α such that A is solvable by means of the logic $L_{\omega\alpha}$ ".

Ma la decisione se una data formula fosse ordinale o meno non rientrava in quest'ultimo tipo di processi. Il quesito di Turing, a cui sperava di dare risposta nella sua tesi, in modo da “tenere sotto controllo l'incompletezza di Gödel” [Hodges, 1985, pp.191 trad. it], era proprio la possibilità di decidere se le formule in questione fossero ordinali o meno. In caso positivo, “esisterebbe un metodo in cui l'aritmetica è completa e tutto potrebbe essere provato a partire da assiomi” [Hodges, 1985, pp.191 trad. it].

Il tentativo di Turing si basa, come in *On Computable Numbers*, sul rifarsi alla dimensione della mente e sull'invenzione di una macchina. Secondo Turing, “il ragionamento matematico può essere considerato in modo schematico come l'esercizio di una combinazione di due facoltà”. Queste facoltà sono “l'intuizione” (*intuition*) e “l'ingegnosità” (*ingenuity*).

La prima “consiste nel formulare giudizi spontanei che non sono il risultato di catene di ragionamenti consapevoli”, come ad esempio il giudizio per cui tutti i numeri interi positivi sono unicamente fattorizzabili in numeri primi. “Questi giudizi sono spesso, ma non sempre, corretti”. Turing non approfondisce in cosa consista questa facoltà e dichiara: “Non tenterò di spiegare più esplicitamente questa idea di 'intuizione’”.

All'ingegnosità spetta il compito di “assistere l'intuizione attraverso un'adeguata disposizione delle proposizioni e, forse, mediante figure geometriche o disegni. L'obiettivo è che, quando queste disposizioni sono realmente ben organizzate, la validità dei passi intuitivi richiesti non possa essere seriamente messa in discussione” [Turing, 1939, pp. 214-215]¹⁹⁸.

Il processo con cui si decide se una formula sia o meno ordinale è rappresentato da Turing come *intuizione*. “In una «logica ordinale completa» dovrebbe dunque essere possibile provare un qualsiasi teorema dell'aritmetica mediante una combinazione di ragionamento meccanico e di passi «intuitivi».” [Hodges, 1985, p.191 trad. it].

Si riproduce, nella distinzione non specificata tra intuizione e ingegno in relazione ai problemi di logica ordinale, la stessa contraddizione presente nel suo lavoro sull'*Eintscheidungsproblem*: alla formalizzazione, alla meccanizzazione totale, sfugge sempre

¹⁹⁸ “Mathematical reasoning may be regarded rather schematically as the exercise of a combination of two faculties J, which we may call intuition and ingenuity. The activity of the intuition consists in making spontaneous judgments which are not the result of conscious trains of reasoning. These judgments are often but by no means invariably correct (leaving aside the question what is meant by “correct”). Often it is possible to find some other way of verifying the correctness of an intuitive judgment. We may, for instance, judge that all positive integers are uniquely factorizable into primes; a detailed mathematical argument leads to the same result. This argument will also involve intuitive judgments, but they will be less open to criticism than the original judgment about factorization. I shall not attempt to explain this idea of “intuition” any more explicitly. The exercise of ingenuity in mathematics consists in aiding the intuition through suitable arrangements of propositions, and perhaps geometrical figures or drawings. It is intended that when these are really well arranged the validity of the intuitive steps which are required cannot seriously be doubted”.

qualcosa. Nel corso di questo lavoro mostreremo le conseguenze di tale irriducibilità rispetto alla riduzione sintattica totale di insiemi formali di simboli. Una volta che in essa confluiranno la teoria statistica dell'informazione e un approfondimento dell'analogia con le facoltà e le caratteristiche del cervello, ciò provocherà un mutamento paradigmatico nelle forme di computazione e di elaborazione digitale; il primo esempio di questo mutamento sarà fornito dallo stesso Turing, come vedremo più avanti.

Come notato sopra a proposito dell'ontologia algoritmica, la macchina di Turing, nella misura in cui rende rigorosa la procedura di computazione, rimane *inerte* senza l'interazione con un essere umano che ne definisca imperativamente scopo e condizioni: "La Macchina di Turing presuppone essenzialmente la presenza di un essere umano come parte del sistema: è l'essere umano a porre le domande, fornire le risorse materiali e interpretare le risposte"[Dodig-Crnkovic, 2011]¹⁹⁹.

Nel caso della procedura indicata per controllare l'incompletezza- sebbene Turing non approfondisca questo punto – il semplice ragionamento meccanico non è sufficiente per decidere se una formula vada considerata come ordinale. Vi è uno scarto. Qui entra in gioco la macchina escogitata da Turing nel suo lavoro dottorale: la macchina *oracolo*, la *o-machine*.

Scriva Turing:

Supponiamo di essere dotati di un mezzo non specificato per risolvere problemi di teoria dei numeri, una sorta di oracolo. Non approfondiremo la natura di questo oracolo, se non per dire che non può essere una macchina". Con l'ausilio di questo oracolo, sarebbe possibile "costruire un nuovo tipo di macchina (chiamiamole *o-machine*.), aventi come uno dei loro processi fondamentali la risoluzione di un dato problema di teoria dei numeri [Turing, 1939, p.172-173 trad. mia]²⁰⁰.

L'oracolo interverrebbe a suggerire al meccanismo computazionale della macchina come risolvere il problema²⁰¹.

¹⁹⁹ "The Turing Machine essentially presupposes a human as a part of a system-the human is the one who poses the questions, provides material resources and interprets the answers".

²⁰⁰ "Let us suppose that we are supplied with some unspecified means of solving number-theoretic problems; a kind of oracle as it were. We shall not go any further into the nature of this oracle apart from saying that it cannot be a machine. With the help of the oracle we could form a new kind of machine (call them *o-machines*), having as one of its fundamental processes that of solving a given number-theoretic problem. More definitely these machines are to behave in this way".

²⁰¹ "Qui, malgrado la reticenza di Turing a instaurare un nesso chiaro tra le facoltà necessarie al ragionamento matematico per risolvere i problemi, tra cui quelli della logica ordinale, e l'utilizzo di quest'ultima per dare forma specifica a questa distinzione, il parallelismo tra intuizione e facoltà della mente sembra emergere. Cfr. Hodges, 1985, p.192 trad. it..

L'oracolo è concepito come un sistema *black box* che amplia la *o-machine*, al quale altrimenti sarebbe una normale Macchina di Turing. "Le loro mosse sono determinate, come di consueto, da una tabella, ad eccezione delle mosse dalla configurazione interna (*o*)" [Turing, 1939, p.173 trad. mia]²⁰².

Le operazioni della *o-machine* sarebbero dunque le seguenti: ricevere in ingresso un programma codificato sotto forma di una sequenza di 1 e 0 (quindi digitale); elaborare il programma applicando passo dopo passo le operazioni di base; e, nei casi richiesti, quando si trova in una configurazione particolare (*o*) passare i dati all'oracolo, che restituisce immediatamente l'informazione richiesta [Copeland, Proudfoot, 1999].

Turing, facendo riferimento alla definizione di macchina *circle-free* analizzata e al problema del *printing* in *On Computable Numbers*, mostra che non esiste un metodo generale per determinare se un' *o-macchina* è (*o*)-*circle free*, ossia se non entrerà mai in un ciclo infinito durante il calcolo. Se esistesse un' *o-macchina* capace di stabilire per qualsiasi altra se è (*o*)-*circle free*, si genererebbe una contraddizione logica: sarebbe possibile costruire un' *o-macchina o-soddisfacente* che stampa un output contraddittorio, in cui una cifra dovrebbe essere contemporaneamente uguale e diversa da sé stessa. Questo dimostra l'impossibilità di determinare in modo generale se un' *o-macchina* è (*o*)-*circle free*. [Cfr. Turing, 1936, p.173]

Questo per Turing rappresenta "la dimostrazione del fatto che i problemi relativi alla *circle freedom* delle *o-macchine* non sono problemi di teoria dei numeri" [Turing, 1939, p.173 trad. mia]²⁰³.

Non c'è quindi un metodo algoritmico per stabilire i problemi posti all'oracolo, e cioè se una determinata formula appartiene alla classe delle formule ordinali. Questa decisione non è una decisione di teoria dei numeri, non è un processo meccanizzabile. "Se il nostro 'oracolo' non ci dice se una data affermazione di teoria dei numeri è vera, ma se una certa formula è una formula ordinale, l'argomento rimane valido e scopriamo che esistono classi di problemi che non possono essere risolti da un processo uniforme, nemmeno con l'aiuto di questo oracolo" [Turing, 1939, p. 200 trad. mia]²⁰⁴.

²⁰² "The moves of the machine are determined as usual by a table except in the case of moves from a certain internal configuration *o*"

²⁰³ "This completes the proof that problems of *circle freedom* of *o-machines* are not number-theoretic".

²⁰⁴ "If our "oracle" tells us, not whether any given number theoretic statement is true, but whether a given formula is an ordinal formula, the argument still applies, and we find that there are classes of problem which cannot be solved by a uniform process even with the help of this oracle".

La risoluzione dei problemi di logica ordinale non può eccedere rispetto alla classe di problemi che appartengono a quella di teoria dei numeri. Il sistema di inferenza non può essere meccanizzato: "Il teorema di Gödel dimostra che un tale sistema non può essere interamente meccanico" [Turing, 1939, p. 200 trad. mia]²⁰⁵.

Quanto accadeva nel lavoro sul problema della decisione, e cioè lì emergere dalla macchina di una dimensione non meccanica (in cui, come abbiamo visto, vi è un'intrinseca interazione con la dimensione umana), così avviene nel ricorso all'oracolo. Seguendo gli assunti fondamentali del programma di Hilbert (e, più indietro, di quello logicista di Russell e prima ancora di Frege), "Le relazioni di dipendenza fra le formule [...] sono così pienamente «visibili»: le loro proprietà e caratteristiche possono, o almeno così si riteneva, essere pienamente trasparenti sulla base delle relazioni puramente sintattiche e strutturali fra le forme delle stringhe di simboli" [Berto, 2009, pp.48-49].

La *o*-macchina, potenziamento della Macchina di Turing, ha invece bisogno di un sistema a *scatola chiusa*, una *black box* per poter risolvere il problema – non meccanizzabile- di decidere se una determinata formula sia ordinale o meno.

²⁰⁵ "We might hope to obtain some intellectually satisfying system of logical inference (for the proof of number-theoretic theorems) with some ordinal logic. Godel's theorem shows that such a system cannot be wholly mechanical". La trattazione di Turing intorno alle *o*-machines aprirebbe all'idea che queste macchine potrebbero realizzare l'ipercomputazione (*Hypercomputation*), quel tipo di computazione che fornirebbe risultati su funzioni non computabili secondo la Tesi di Church-Turing, in quanto la presenza dell'oracolo, sebbene Turing non specifichi mai come realizzarla, permetterebbe di superare i limiti di una macchina di Turing Universale. Questo permetterebbe di risolvere problemi dimostrati incomputabili, come l'*halting problem*, o di realizzare un *debugger* perfetto, capace di identificare tutti i bug, inclusi quelli che provocano loop infiniti [Copeland, 1999]. Sull'ipercomputazione si veda Piccinini, 2025. In questo ambito, significative sono le critiche di Davis [Davis, 2006], provenienti da chi, partendo dal problema del printing - dimostrato indecidibile da Turing nel 1937 - ha messo in luce l'incomputabilità del problema dell'arresto. Nella pratica l'ipercomputazione non esisterebbe in quanto non potremmo mai verificare se un dispositivo stia davvero producendo risultati non calcolabili secondo la tesi di Church-Turing, dato che abbiamo accesso solo a dati finiti. Inoltre, le teorie sull'ipercomputazione si baserebbero su teorie fisiche non verificate dato che non esiste alcun meccanismo fisico noto in grado di generare funzioni non calcolabili in modo dimostrabile. Il lavoro di Turing rappresenta inoltre un ponte tra le teorie gödeliane sui limiti della computabilità e quelle della cognizione. A questo proposito, malgrado la differenza dagli esiti del matematico inglese siano molto lontani da queste, è interessante citare la posizioni di J.R. Lucas, che nel suo articolo *Minds, Machines and Gödel* [Lucas, 1961]. Il teorema di incompletezza di Gödel, secondo Lucas, evidenzia una differenza fondamentale tra la mente umana e le macchine. Poiché ogni macchina è un'istanza concreta di un sistema formale coerente, esisteranno sempre enunciati veri che essa non può dimostrare. Al contrario, la mente umana è in grado di riconoscere la verità di tali enunciati gödeliani, mostrando una capacità che trascende i limiti imposti dai sistemi meccanici. Di conseguenza, la mente umana non può essere completamente descritta né replicata da alcuna macchina algoritmica, poiché essa trascende qualsiasi sistema formale che tenti di rappresentarla. Il matematico, filosofo, computer scientist e scienziato cognitivo Douglas Hofstadter, in *Gödel, Escher, Bach: un'Eterna Ghirlanda Brillante* [Hofstadter, 1979], testo vincitore del Premio Pulitzer per la saggistica nel 1980, utilizza il teorema di Gödel per esplorare la natura della coscienza e dell'autoreferenza. Le strutture autoreferenziali degli enunciati gödeliani diventano un modello per comprendere come la mente umana possa emergere da livelli gerarchici aggrovigliati di processi ricorsivi. Analogamente, nelle opere di Bach ed Escher, Hofstadter identifica schemi di auto-riflessione e loop infiniti che suggeriscono come la coscienza possa derivare da un sistema complesso di auto-interpretazione, transcendendo i limiti del rigido formalismo dei sistemi meccanici.

L'idea fondamentale della Macchina Universale di Turing - analoga alla gödelizzazione - consiste nel rendere omogenei numeri e operazioni sui numeri, input e regole di manipolazione, ossia dati e programmi: un principio destinato a rivelarsi decisivo per lo sviluppo del calcolatore moderno, della scienza cognitiva e dell'intelligenza artificiale. Essa, tuttavia, sembra lasciarsi sfuggire qualcosa.

Anche nella sua versione "*augmented*" [Copeland, Proudfoot, 1999], la macchina oracolo, la Macchina di Turing sembra di aver bisogno del riferimento a una dimensione ulteriore.

C'è comunque in questo lavoro di Turing la stessa ambiguità riscontrabile in *Computable Numbers*: la mente viene meccanizzata, ma al tempo stesso si accenna all'esistenza di qualcosa al di là della meccanizzazione. Qual è il significato di tutto questo per la mente umana? A questo stadio Alan non ha ancora le idee chiare [Hodges, 1985, p.192 trad. it.].

La stessa ambiguità, lo stesso scarto tra le possibilità computazionali delle macchine di Turing e ciò che vi sfugge, riemerge quando le sue idee, divenute decisamente più chiare grazie al lavoro nella crittografia di guerra a Bletchley Park e al contributo alla costruzione di macchine - come l'ACE e la Manchester Mark I, di cui vedremo le caratteristiche - trovarono concreta realizzazione. Tale scarto non riguarda soltanto la computazione, il software o il supporto materiale nei calcolatori, l'hardware, ma si estende anche al *wetware*: quella sostanza dalla consistenza di *porridge* freddo che è il cervello umano. [Turing, 1952, trad. mia]²⁰⁶.

Turing in un lavoro pionieristico ma a lungo misconosciuto, *Intelligent Machinery* del 1948 [Turing, 1948], scritto mentre lavorava sulle cellule nervose artificiali e lavorava come *chief architect* per il progetto dell'*Automatic Computing Engine* (ACE) presso il *National Physical Laboratory* (NPL), parla esplicitamente di un "residuo" [*residue*]: " Il nostro compito è scoprire la natura di questo «residuo» e cercare di «copiarlo» nelle macchine" [Turing, 1948, p. 117 trad. it.]²⁰⁷.

Il lavoro di Turing non fu mai pubblicato perché il direttore dell'NPL, Sir Charles Darwin, nipote del ben più celebre, e scientificamente più lungimirante, naturalista, ne sconsigliò la pubblicazione bollandolo come "tema scolastico" (*schoolboy's essay*) [Copeland, 2004b, p.401]

In cosa consiste il residuo di cui parla il matematico inglese e quale legame ha con la capacità computazionale del cervello? La risposta a questa domanda consente di riconoscere in

²⁰⁶ "The brain has the consistency of cold porridge".

²⁰⁷ Il tema è sottolineato con forza in Marconi, 2025.

Turing uno dei precursori [Copeland, Proudfoot 1996 e 1999; Copeland 2004cc, pp. 403-409; Dewhurst, 2019] del paradigma connessionista - solitamente fatto risalire, nella sua forma moderna, al *Perceptron* di Rosenblatt [Rosenblatt, 1958].

Il *connessionismo*, come analizzeremo in dettaglio più avanti, è quel movimento della scienza cognitiva che mira a spiegare le capacità intellettuali e il comportamento umano attraverso reti neurali artificiali distribuite in parallelo, composte da unità simili a neuroni, in cui l'apprendimento è mediato da cambiamenti nella forza delle connessioni tra gli elementi, modellando così il pensiero e i processi cognitivi [Arbib, 2003, p. 12; Buckner & Garson, 2019].

Turing infatti, dopo aver fornito una tassonomia di macchine sul modello della Macchina di Turing e delle loro caratteristiche²⁰⁸, ne propone un tipo ulteriore. "Potremmo invece

²⁰⁸ Spicca tra le varie macchine descritte da Turing (1948, pp. 93-97, trad. it.) la Macchina di Carta (*Paper Machine*) (ivi, p. 97, trad. it.), che implementa una Macchina Universale in un essere umano dotato di carta e matita. Questo ibrido umano-algoritmico dimostra il legame necessario tra l'essere umano (in questo caso il matematico) e la Macchina di Turing, affinché quest'ultima non rimanga inerte. Le diverse macchine proposte da Turing sono riassunte nella tabella [Tabella 1] sottostante:

Tipo di Macchina	Definizione	Caratteristiche	Esempio Moderno
Macchine Calcolatrici Logiche, <i>Logical Computing Machines</i> (MCL) Universalità: No	Macchine astratte con memoria infinita disposta su un nastro suddiviso in quadrati.	Quelle della macchina di Turing. Nastro su cui una testina legge e scrive simboli	Macchina di Turing teorica
Macchine Calcolatrici Logiche Universali, <i>Universal Logical Computing Machines</i> (MCLU) Universalità: Si	Generalizzazione delle MCL in grado di simulare qualsiasi altra MCL	Legge la descrizione della macchina da un nastro e la esegue.	Computer programmabile
Macchine Calcolatrici Pratiche, <i>Practical Computing Machines</i> (MCP) Universalità: No	Macchine progettate per il calcolo pratico con vincoli di tempo e memoria.	Usano memoria organizzata in modo più efficiente rispetto al nastro infinito delle L.C.M.s. Accesso più rapido ai dati.	Mainframe, PC
Macchine Calcolatrici Pratiche Universali, <i>Universal Practical Computing Machines</i> (MCPU) Universalità: Si	Macchine pratiche con le stesse proprietà delle MCLU quindi programmabili per qualsiasi compito computabile.	Un singolo computer può eseguire diversi compiti grazie alla programmazione.	Supercomputer, server cloud
Macchine di Carta, <i>Paper Machine</i> Universalità: Si	Un essere umano che segue rigidamente un insieme di istruzioni scritte può funzionare come una macchina di Turing.	Una persona con carta, matita e gomma simula una macchina calcolatrice.	Matematico che segue un algoritmo manualmente
Macchine Parzialmente Casuali, <i>Partially Random Machines</i> Universalità: Dipende	Macchine che includono scelte casuali in determinati punti dell'operazione.	Il comportamento della macchina non è deterministico, ma include un elemento di casualità.	Algoritmi Monte Carlo

considerare cosa accade quando costruiamo una macchina in modo relativamente non sistematico, utilizzando una sorta di componenti standard” [Turing 1948, in Copeland, 2004, p.416]: le “macchine non organizzate” [Turing, 1948 p.98 trad. it.]²⁰⁹.

Questo tipo di macchine sono quelle composte da N unità, ognuna con due ingressi e un’uscita, che possono assumere come valore 0 o 1. Le connessioni tra queste unità sono casuali. “Possiamo immaginare che per ogni intero r , $1 \leq r \leq N$, due numeri $i(r)$ e $j(r)$ siano scelti a caso da $1...N$ e che gli ingressi dell’unità r siano connessi alle uscite di $i(r)$ e di $j(r)$. Tutte le unità sono connesse a un’unità centrale di sincronizzazione, che emette impulsi a intervalli più o meno uguali di tempo” [Turing, 1948 p.98 trad. it.]

Il comportamento, lo “stato” delle unità sono determinati, in ogni momento, da una regola secondo cui lo stato attuale è calcolato sottraendo a 1 il prodotto degli stati delle due unità collegate nell’istante precedente. “Lo stato è determinato da questa regola: sottrarre a 1 la quantità ottenuta moltiplicando loro gli stati al momento precedente delle unità da cui provengono gli stimoli di ingresso” [Turing, 1948 p.98 trad. it.].

Turing qui stabilisce quelle che nel linguaggio del connessionismo contemporaneo possiamo chiamare una *regola di propagazione* (1) e una *regola di attivazione* (2):

- (1) L’input netto nell’unità r al momento m , $\text{net}(r,m)$ è il prodotto dello stato di $i(r)$ al momento $m-1$ e dello stato di $j(r)$ al momento $m-1$
- (2) Lo stato dell’unità r al momento m è dato a $1 - \text{net}(r,m)$ [Copeland, Proudfoot, 1996]

Se entrambi gli ingressi sono 1, allora il prodotto è 1, quindi $1 - 1 = 0 \rightarrow$ lo stato diventa 0; se almeno uno degli ingressi è 0, il prodotto è 0, quindi $1 - 0 = 1 \rightarrow$ lo stato diventa 1. La macchina di tipo A opera secondo una logica NAND, un’operazione booleana (*NOT AND*) che restituisce falso solo quando entrambe le variabili di ingresso sono vere, e vero in tutti gli altri casi, fungendo sia da operatore base universale che da porta logica fondamentale nei circuiti digitali.

Le reti di unità connesse secondo queste modalità sono quelle che Turing definisce le “macchine di tipo A” (*A-type unorganised machines*) [Turing, 1948 p.99 trad. it.]. Le macchine di

Macchine Apparentemente Parzialmente Casuali, <i>Apparently Partially Random</i> <i>Machines</i>	Macchine deterministiche che sembrano comportarsi in modo casuale.	Possono usare numeri deterministici (es. i decimali di π) per generare scelte apparentemente casuali.	Algoritmi crittografici
Universalità: Dipende			

²⁰⁹ “Unorganised machines” nell’originale.

tipo A, “sono forse il più semplice modello di un sistema nervoso con una distribuzione casuale di neuroni” [Turing, 1948 p.100 trad. it.].

Tabella. 2 Comportamento di un neurone NAND [Copeland, 2004cc, p.404]

Input-1	Input-2	Output
1	1	0
1	1	1
0	0	1
0	0	1

Le connessioni delle macchine di tipo A, la cui complessità di comportamento cresce in relazione al numero di unità, possono essere indicate con $\longrightarrow$. Turing introduce un secondo tipo di macchine non organizzate, quelle in cui “rimpiazziamo la connessione $\longrightarrow$ con $\rightarrow \square \rightarrow$ ” [Turing, 1948, p.100 trad. it.].

Questo tipo di macchine sono le “macchine non organizzate di tipo B” [*B-type unorganised machines*]. La macchina di tipo B “Consiste di neuroni artificiali interconnessi [...] e di modificatori di connessione, rappresentati come quadrati” [Copeland, 2004cc, p.403 trad. mia]²¹⁰.

Le macchine di tipo B sono quelle macchine non organizzate che sono passibili di addestramento interferente [*interfering training*]. “Molte macchine non organizzate hanno configurazioni tali che, una volta raggiunta una di queste, e se l’interferenza è da quel momento in avanti opportunamente limitata, la macchina si comporta come se fosse organizzata in vista di uno scopo definito” [Turing, 1948, p.106 trad. it.].

A questa possibilità di controllo delle condizioni iniziali Turing attribuiva la massima importanza. Controllare l’attivazione della macchina significa per il matematico inglese far sì che mediante la modificazione di questi, la macchina si organizzi. “Il processo di attivazione delle condizioni iniziali, per far sì che la macchina esegua qualche compito particolarmente utile, può essere chiamato «organizzazione della macchina». L’organizzazione è dunque una

²¹⁰ “In ‘Intelligent Machinery’ Turing introduced the type of neural network that he called a ‘B-type unorganised machine’. A B-type consists of interconnected artificial neurons [...] as circles, and connection-modifiers, depicted as squares

forma di modificazione” [Turing, 1948, p.108 trad. it.]. Questo processo di organizzazione è reso possibile dai modificatori.

Tutti i modificatori di connessione possiedono due fibre di addestramento. L'invio di un impulso a una delle due fibre attiva la modalità passaggio (*pass mode*), in cui l'input-0 o 1-viene trasmesso inalterato come output. Un impulso sull'altra fibra attiva invece la modalità interruzione (*interrupt mode*), forzando l'output a 1 indipendentemente dall'input. In questa modalità, il modificatore distrugge tutte le informazioni che tentano di attraversarlo, bloccando il flusso di dati. [Copleand, 1999].

La connessione, attraverso il modificatore “□”, può avere tre effetti che dipendono dallo stato: “Può (a) passare tutti i segnali scambiando tra loro 0 e 1, oppure (b) può convertire tutti i segnali nel segnale costante 1, o ancora (c) può agire come in (a) e in (b) a momenti alternati. (Il caso (c) ha due sottocasi). Quale di questi casi si applichi dipende dalle condizioni iniziali. C'è poi un ritardo di due momenti nel passare attraverso la connessione $\rightarrow \square \rightarrow$.” [Turing, 1948, p.100 trad. it.].

Tabella 3: il comportamento di una Macchina non organizzata di tipo B

Input-1	Input-2	Output	Caso
0	0	{0,1}	(a) Scambio $0 \leftrightarrow 1$ o (b) Tutto 1
0	1	1	(b) Tutto 1
1	0	1	(b) Tutto 1
1	1	{0,1}	(a) Scambio $0 \leftrightarrow 1$ o (c) alternanza

Una macchina di tipo B “può essere trasformata, con opportune condizioni iniziali, in macchina universale con memoria fissata” [Turing, 1948, p.106 trad. it.]. Queste macchine attraverso l'addestramento, “con opportune condizioni iniziali, esse sono in grado di svolgere

qualsiasi lavoro, date un tempo sufficiente e un numero adeguati di unità” [Turing, 1948, p.106 trad. it.]²¹¹.

L’analisi di queste macchine non ha per Turing soltanto uno scopo legato allo sviluppo delle macchine calcolatrici come l’*Automatic Computing Engine*. In una lettera al cibernetico britannico William Ross Ashby²¹² (1903-1972), Turing afferma esplicitamente: “Nel lavorare all’ACE, sono più interessato alla possibilità di produrre modelli dell’azione del cervello che alle applicazioni pratiche nel calcolo” [Turing, 19 novembre 1946, lettera a W.R. Ashby; Copeland, Proudfoot, 1996 trad. mia]²¹³. La macchina non organizzata fornisce infatti un modello esplicativo della struttura cerebrale umana: “L’immagine della corteccia come macchina non organizzata è molto soddisfacente dal punto di vista evolutivo e genetico” [Turing, 1948, p.106 trad. it.].

Turing riteneva infatti che, accanto ai centri finalizzati del sistema nervoso, come quelli percettivi o quelli del sistema nervoso autonomo, “ci siano ampie zone del cervello, soprattutto nella corteccia, la cui funzione è in larga misura indeterminata”. L’organizzazione specializzata che queste zone - indeterminate nel neonato - acquisiscono, ad esempio quelle implicate nell’imparare una determinata lingua come il francese o l’inglese, “dipend[ono] dall’addestramento dell’individuo nel corso della fanciullezza”. “Tutto questo suggerisce che la corteccia del neonato sia una macchina non organizzata, che può essere organizzata con un opportuno addestramento tramite interferenza” [Turing, 1948, p. 108 trad. it.].

²¹¹ In realtà, la pretesa di Turing che le macchine non organizzate di tipo B possano svolgere ogni lavoro è infondata. Il problema risiede nei modificatori di connessione, che operano solo in modalità *interrupt* o *interchange*. Questa limitazione impedisce a una macchina di tipo B di eseguire alcune operazioni logiche fondamentali, come la disgiunzione esclusiva (XOR) o la funzione identità, poiché un numero dispari di interscambi non permette di tornare allo stato originale. Una possibile soluzione consiste nell’adozione di modificatori che, invece della modalità *interchange*, abbiano la modalità *pass*, in cui l’output coincide con l’input. Questo permetterebbe alle macchine di tipo B di eseguire ogni compito richiesto, rendendoli effettivamente universali. Inoltre, tali modificatori potrebbero essere costruiti combinando coppie di modificatori *interrupt/interchange*, ottenendo una configurazione che replica la modalità *pass* o *interrupt* a seconda delle necessità. Copeland, 2004cc, p.406.

²¹² Alan Turing ebbe contatti significativi con il movimento cibernetico britannico attraverso il *Ratio Club*, un gruppo interdisciplinare fondato nel 1949 dal neurologo John Bates. Il club riuniva giovani psicologi, fisiologi, matematici e ingegneri per discutere temi legati alla cibernetica, con l’obiettivo di promuovere una comprensione più profonda dei processi di calcolo e delle facoltà mentali umane. Tra i membri più importanti figuravano W. Ross Ashby e W. Grey Walter. Turing fu invitato a partecipare dopo il primo incontro e accettò con entusiasmo, contribuendo alle discussioni con le sue idee innovative sulla computazione e sull’intelligenza delle macchine. Presentò due conferenze: *Educating a Digital Computer*, in cui esplorava la possibilità di macchine capaci di apprendere, e *The Chemical Basis of Morphogenesis* [Turing, 1952b], dove applicava modelli matematici per spiegare la formazione delle strutture biologiche. Il *Ratio Club* si riunì regolarmente fino al 1958. Dewhurst, 2019; Husbands, Hollands, 2008.

²¹³ In working on the ACE I am more interested in the possibility of producing models of the action of the brain than in the practical applications to computing.

Applicare le giuste modificazioni, addestrare la macchina non organizzata – qualora si stesse cercando “di costruire una macchina intelligente e abbiamo deciso di seguire per quanto possibile il modello umano” - attraverso un processo che si attua “simulando l’educazione” [*mimicking education*], dovrebbe allora consentire di trasformare la macchina non organizzata in una macchina universale [Turing, 1948, p.105 trad. it.].

Ma quale implicazione ha, per il cervello, la possibilità di addestramento delle macchine non organizzate? Nel lavoro di Turing si intravede lo stesso scarto tra la dimensione computazionale e un’istanza a essa irriducibile che già si era manifestato nei suoi scritti del 1936 e del 1938. In un certo senso, “era possibile, quindi, che la macchina morbida fosse veramente un computer” [Copeland, 2012, p. 274 trad. it.].

Infatti, se la sua analisi delle macchine di tipo A e di tipo B [fig.12,13] “suggerisce che la corteccia del neonato sia una macchina non organizzata che può essere organizzata tramite opportuno addestramento tramite interferenza”, il risultato potrebbe essere la trasformazione del cervello dell’infante in una macchina universale: “L’organizzazione potrebbe risultare nella trasformazione della macchina in una macchina universale, o qualcosa del genere” [Turing, 1948, p. 108 trad. it.].

Ma che ne sarebbe del cervello dell’adulto organizzato in questo modo e del suo comportamento? Se il cervello “non fosse nient’altro che la macchina universale di Turing”, una volta terminata l’istruzione Turing-equivalente per la quale è stato addestrato, modificato e finalizzato, ossia *programmato*, smetterebbe di funzionare.

La persona “che ha eseguito il programma di istruzioni archiviato nell’equivalente cerebrale del nastro” [Copeland, 2012, p. 274 trad. it.], dopo aver svolto il compito senza alcuna obiezione, cadrebbe in una sorta di coma. Scrive infatti Turing:

Tutto questo suggerisce che la corteccia del neonato sia una macchina non organizzata, che può essere organizzata con un opportuno addestramento tramite interferenza. L’organizzazione potrebbe risultare nella trasformazione della macchina in una macchina universale, o qualcosa del genere. Questo significa che l’adulto obbedirebbe ai comandi dati in un appropriato linguaggio, per quanto complicati fossero; sarebbe privo di buon senso, e obbedirebbe agli ordini più ridicoli senza battere ciglio. *Quando tutti gli ordini impartiti fossero stati eseguiti, cadrebbe in uno stato comatoso* [enfasi mia] o forse continuerebbe a obbedire a qualche ordine permanente, come quello di mangiare. [...] Tuttavia la somiglianza con una macchina universale resta molto forte, e ci suggerisce che il passaggio da un neonato non organizzato a una macchina universale è un passaggio che dovrebbe essere compreso appieno [Turing, 1948, pp.108-109 trad. it.].

Trasformare la macchina non organizzata in una macchina universale, trasformare la corteccia indeterminata in una organizzazione finalizzata, è una forma di “*disciplina*” [Turing, 1948, p.116 trad.it. enfasi mia]. Ma la disciplina, lo sforzo di addestramento necessario all’organizzazione è solo una delle condizioni richieste affinché vi possa essere intelligenza. Da sola non è sufficiente. È necessaria un'altra facoltà: “*l’iniziativa*”. Ed è proprio quest’ultima a costituire il “residuo” dal quale la nostra analisi è partita.

Perché la mente non addestrata del neonato diventi intelligente, dovrà acquisire sia disciplina sia capacità di iniziativa. Finora abbiamo considerato solo la capacità di obbedire: convertire un cervello o una macchina in una macchina universale è la forma più estrema di disciplina. Ma la disciplina da sola non è certamente sufficiente a produrre intelligenza: è richiesta, in aggiunta, anche l’iniziativa (e questa frase dovrà servire da definizione del concetto). Il nostro compito è quello di scoprire la natura di questo «residuo» quale si presenta nell’uomo, e cercare di «copiarlo» nelle macchine [Turing, 1948, p.116 trad.it.].

Fig.12 La macchina non organizzata di tipo A

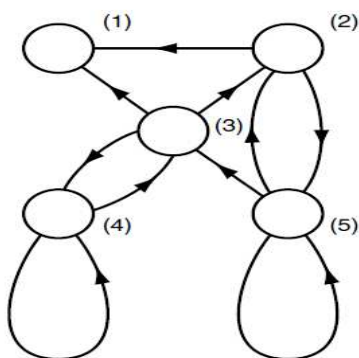
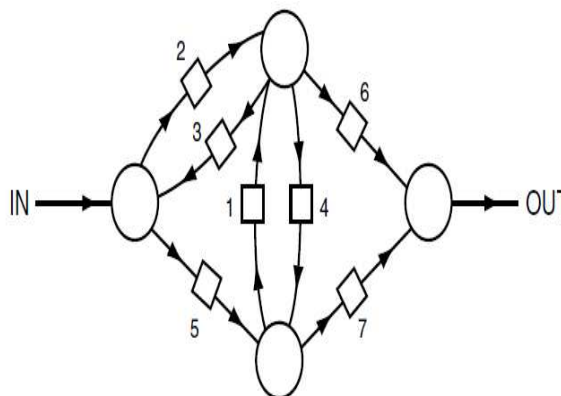


Fig.13 La macchina non organizzata di tipo B [Turing, 1948]



Questa situazione si ripeterebbe anche considerando quella “combinazione”, quell’ibrido tra l’essere umano (il matematico) e l’algoritmo, che consiste nel “produrre l’effetto di una macchina calcolatrice scrivendo un insieme di regole e di procedure, e chiedendo a una persona di eseguirle”, la cosiddetta “macchina di carta” [*paper machine*]. “Una persona fornita

di carta, matita e gomma, e assoggettata a una severa disciplina è in effetti una macchina universale” [Turing, 1948, p. 96 trad. it.].

Anche nel caso dell’istanziamento/implementazione di una macchina universale in un essere umano che svolge un compito Turing-equivalente, vi è una dimensione che sfugge all’addestramento finalizzato allo scopo. Senza questo “residuo”, anche la macchina di carta è destinata a un coma computazionale. Come ovviare a queste difficoltà presenti nelle *paper machines* e nelle macchine non organizzate di tipo A e di tipo B? Come instillare l’iniziativa in macchine che sono costituite intorno all’altra facoltà? A questo problema Turing propone due modalità di risoluzione, che toccano le sue idee sul *machine learning*.

Un “metodo diretto” consiste nell’installare su una macchina del tipo MCLU – la macchina di Turing universale – un programma capace di eseguire qualsiasi lavoro realizzabile in linea di principio, consentendo alla macchina di prendere progressivamente sempre più «scelte» e «decisioni» su cosa fare. Quando il suo comportamento si baserà su principi generali sufficientemente astratti, non richiederà più interferenze esterne e potrà essere considerata “adulta” [Turing, 1948, p. 117, trad. it.].

“Nell’altro metodo si parte con una macchina non organizzata e si cerca di incorporarvi disciplina e iniziativa nello stesso tempo: invece di cercare di organizzarla solo come macchina universale, organizzarla anche in vista delle manifestazioni di iniziativa” [Turing, 1948, p. 117, trad. it.]. Turing tentò un approccio di questo genere facendo ricorso alle “macchine non organizzate di tipo P”, nelle quali confidava come sviluppo delle MCLU capace di superare i limiti delle macchine di carta. Una macchina di tipo P “con memoria esterna ha già molta più «organizzazione» di una macchina di tipo A” [Turing, 1948, p. 114, trad. it.] e, a differenza delle macchine non organizzate di tipo A e B, non si presentava come un modello di rete neurale. Essa, piuttosto, “può essere considerata una MCL senza nastro, e con una descrizione largamente incompleta” [Turing, 1948, p. 111, trad. it.].

L’obiettivo di Turing era trasformare queste macchine, simili, sebbene fornite di tavole di istruzioni incomplete, alle macchine computazionali semplici descritte in *On Computable Numbers*, in macchine universali, attraverso un addestramento secondo un “sistema piacere-dolore” [Turing, 1948, p. 110, trad. it.] simile al modo in cui viene educato un bambino. “L’educazione di un bambino dipende in larga misura da un sistema di premi e punizioni, e questo suggerisce che dovrebbe essere possibile realizzare l’organizzazione con solo due stimoli di interferenza, uno per «piacere» (P) o «premio» e uno per «dolore» (D) o «punizione»” [Turing, 1948, p. 110, trad. it.].

A partire da questa idea, l'addestramento da parte di un insegnante [*teacher*] di una macchina di tipo P avviene attraverso due tipi di stimoli sensoriali trasmessi lungo linee di ingresso: quelli di piacere e dolore (P e D), che modificano direttamente la tabella della macchina, e quelli sensoriali (S), che invece forniscono informazioni senza alterarne lo stato. Le linee sensoriali, che trasportano segnali binari (0 o 1), sono il modo in cui l'insegnante comunica "in modo non emotivo" con la macchina, diversamente dagli stimoli di piacere e dolore.

Inizialmente, la tabella delle transizioni della macchina è "largamente incompleta", e quando deve prendere una decisione lo fa casualmente. Se riceve uno stimolo di piacere, le scelte provvisorie vengono consolidate; se riceve uno stimolo di dolore, vengono cancellate, riportando la macchina a uno stato incerto. Questo processo guida l'apprendimento, permettendo alla macchina di stabilizzarsi su un comportamento desiderato senza bisogno di programmazione esplicita [Turing, 1948, p. 110, trad. it.].

"La tecnica effettiva attraverso cui l'«organizzazione» della macchina di tipo P è stata portata a compimento è forse un po' deludente, perché non è sufficientemente analoga al tipo di processo con cui si insegnerebbe davvero a un fanciullo." [Turing, 1948, p. 110, trad. it.]. Il metodo che utilizzò per addestrare la macchina con memoria esterna dipendeva troppo dalla disponibilità di "considerevole intelligenza" da parte del formatore. [Copeland, Proudfoot, 1996].

Malgrado queste limitazioni, le macchine non organizzate discusse in *Intelligent Machinery* – pur prive dell'elemento decisivo che segnerà il successo del connessionismo contemporaneo, ossia la ponderazione dei pesi nell'addestramento e la regola matematica della retropropagazione dell'errore, che Turing non prese in considerazione – rappresentano uno dei primi tentativi di concepire macchine computazionali basate su elementi semplici, simili a neuroni, connessi in reti in larga parte casuali. In questo senso, esse possono essere considerate antesignane delle moderne architetture di calcolo che, negli anni Ottanta, avrebbero inaugurato la *Golden Age* del connessionismo. Ciò non solo perché, come abbiamo visto, esse risultano "più fedeli a ciò che sappiamo della fisiologia del cervello", ma anche per una ragione di carattere cognitivo: le macchine protoconnessioniste di Turing, al pari delle odierne reti neurali artificiali, riflettono una caratteristica cruciale del cervello – che prenderemo in esame più avanti – ovvero la gestione dell'errore e la *graceful degradation* (degradazione graduale).

A differenza dei calcolatori tradizionali, che in caso di errore subiscono fallimenti catastrofici [*catastrophic failure*], tali sistemi sono in grado di continuare a funzionare nonostante danni e/o fluttuazioni dell'input abituali [Buckner, Garson, 2019, p. 79]²¹⁴.

Torneremo più avanti su questo punto cruciale, mostrando come la gestione dell'errore e l'integrazione della teoria dell'informazione nel cuore stesso della logica computazionale abbiano rappresentato uno dei luoghi genetici del cambio di paradigma, in cui Turing si colloca come snodo centrale. Da un lato, le sue macchine hanno fornito il modello e la logica per concepire la mente come manipolatrice di simboli attraverso regole implementabile nel calcolatore; dall'altro, i suoi tentativi contengono già, *in nuce*, il passaggio verso forme di elaborazione dell'informazione differenti, più vicine a una descrizione del sistema e delle funzioni del cervello.

4. Imitare quel ronzio nella mia testa

Turing aveva in mente le macchine di tipo P quando, nel 1950, in *Computing Machinery and Intelligence* [Turing, 1950], affrontando la questione delle possibilità di superamento del celebre *Imitation Game*, scrisse: "Ho fatto alcuni esperimenti con una simile macchina-bambino e sono riuscito a insegnarle alcune cose, ma il metodo d'insegnamento era troppo poco ortodosso perché gli esperimenti potessero venir considerati veramente riusciti" [Turing, 1950, pp. 152-153, trad. it.].

Una conferma di questa idea si trova nel fatto che al termine *Intelligent Machinery*, Turing propone una versione embrionale del *Gioco dell'imitazione* [Copeland, 2000].

È possibile fare un piccolo esperimento in questa direzione, anche allo stato presente delle nostre conoscenze. Non è difficile preparare una macchina di carta che giochi non troppo male a scacchi. Si prendano ora tre soggetti A, B, C; A e C devono essere mediocri giocatori di scacchi, B l'operatore che manovra la macchina di carta. (Affinché riesca a lavorare a buon ritmo è consigliabile che sia allo stesso tempo un matematico e un giocatore di scacchi). Si usano due stanze separate, con accorgimenti del caso

²¹⁴ "Of course one argument was bound to be that neural nets were more faithful to what we know about the physiology of the brain. But there were arguments at the cognitive level as well. For example, neural nets showed a much more robust and realistic response to damage and to noisy input. The loss or random modification of units in a classical computer typically causes catastrophic failure (or none at all, as in the loss of an unused memory location). However, when units of a connectionist model trained with backpropagation are removed, or their connections randomly modified, the network usually continues to function appropriately, albeit less accurately. This "graceful degradation" of function is arguably an essential feature of the brain, which routinely suffers damage and/ or random fluctuations in the input".

per comunicare le mosse, e si giocano due partite: una tra C e A, e una tra C e la macchina di carta. C può trovare notevoli difficoltà a dire contro chi sta giocando. (Questa è una forma idealizzata di un esperimento che io ho in effetti compiuto) [Turing, 1948, p.119-120 trad. it.].

In cosa consiste l'*Imitation Game*? Il *Test di Turing*"²¹⁵ Il matematico inglese - dopo aver affermato come la domanda "Possono pensare le macchine?" [Turing, 1950, p.121 trad. it. 1994] sia insensata e che debba essere scartata - "Credo che la domanda iniziale: «Possono pensare le macchine?», sia troppo priva di senso per meritare una discussione per stabilire se le macchine siano intelligenti" [Turing, 1950, p.133 trad. it. 1994]- propone un gioco, il *Gioco dell'imitazione*.

Ipirandosi a un popolare gioco di ruolo dell'epoca - che coinvolge tre partecipanti, A (uomo), B (donna) e C (giudice), che viene separato dagli altri due e deve determinare chi è l'uomo e chi la donna attraverso semplici domande e risposte scritte- Turing riformula la domanda sul pensiero delle macchine in un'ipotesi scientifica verificabile: se una macchina sostituisce un partecipante nel gioco e ingannasse un giudice che non la distinguesse da un essere umano, allora alla macchina deve essere attribuita l'intelligenza [Turing, 1950, pp.121-123 trad. it. 1994].

Turing non sta proponendo una definizione di intelligenza, o di pensiero, [Copeland, 2000], né tantomeno illustrando un criterio comportamentistico o operativo di intelligenza [Moor, 2001], ma "il nostro potenziale interesse per il gioco dell'imitazione è suscitato [...] dal

²¹⁵ Indipendentemente dall'analisi svolta qui del Test di Turing, è necessario riconoscere che il suo superamento "è stato perseguito prevalentemente in manifestazioni socio-culturali, come il *Loebner Prize*" [Cordeschi, Tamburrini, 2015]. Quest'ultimo è un premio annuale istituito nel 1991 dal filantropo Hugh Loebner per il miglior chatbot in una competizione basata sul *Turing Test*. Finanziato dallo stesso Loebner, il concorso si è svolto fino al 2019 in diverse sedi, tra cui l'Università di Reading, e prevedeva premi in denaro per i programmi più convincenti. Il riconoscimento massimo di 100.000 dollari era riservato alla prima IA capace di superare effettivamente il test. Per un'analisi dal punto di vista di filosofi che hanno fatto parte della giuria di una particolare edizione del premio, si veda Floridi et al., 2009. Per una panoramica più ampia sul *Turing Test*, si rimanda a Oppy, Dowe 2021 e ed. Moor, 2003. Larsson (Larsson 2021), invece, ne analizza le difficoltà come criterio di valutazione dell'intelligenza alla luce delle operazioni di inferenza logica. Non si può non citare, inoltre, data l'influenza culturale dell'*Imitation Game*, il *Test di Voight-Kampff*, basato sulla capacità empatica, elaborato da Philip K. Dick nel romanzo di fantascienza *Do Androids Dream of Electric Sheep?* [Dick, 1968], che ha ispirato il film *Blade Runner* di Ridley Scott (1982). Inoltre, ciascuno di noi, probabilmente più volte al giorno, si sottopone inconsapevolmente a un *Test di Turing*, in cui, paradossalmente, è la macchina a giudicare la nostra umanità: accade quando incontriamo online la familiare schermata di un CAPTCHA (*Completely Automated Public Turing test to tell Computers and Humans Apart*). Sebbene i CAPTCHA siano stati a lungo un compito difficile per le macchine, gli sviluppi recenti nell'intelligenza artificiale hanno reso molti di essi facilmente superabili. Ad esempio, un algoritmo sviluppato dal Politecnico di Zurigo ha mostrato una precisione del 100% nel riconoscere immagini di traffico urbano utilizzate nei reCAPTCHA, mettendo in discussione la loro efficacia come test distintivo tra umani e IA avanzate [Plesner et al., 2024].

fatto che in qualche modo questo test rivela una connessione tra le possibili attività del computer e il nostro concetto ordinario di pensiero" [Moor, 1976 trad. mia]²¹⁶.

Da questo punto di vista, l'*Imitation Game* si configura come un test sulla nostra teoria dei processi che avvengono all'interno della macchina, così come all'interno di un altro essere umano. Ciò lo distingue da un semplice test volto a misurare la *credulità* dell'interrogante [Copeland, 2000].

Anche versione embrionale del Test del 1948 - riguardante le macchine non organizzate di tipo P, quelle macchine logiche a cui stava cercando di instillare il "residuo" proprio del cervello umano intelligente- Turing afferma che:

Quanto siamo disposti a considerare intelligente il comportamento di una qualche entità, dipende tanto dal nostro stato mentale e dalla nostra preparazione quanto dalle proprietà dell'oggetto in esame. Se siamo capaci di spiegare e predire il suo comportamento, o se ci sembra che esso sia poco pianificato, non siamo molto tentati di immaginare un'intelligenza in azione. Perciò è possibile che una persona consideri un oggetto intelligente e un'altra, che ha scoperto le regole del suo comportamento, no [Turing, 1948, p.119 trad. it].

Possiamo a questo punto, ricostruiti gli antecedenti dell'*Imitation Game* nel lavoro del 1948, applicare il "*Principio di Turing*": "Una macchina che, mediante calcolo, imita – o, meglio, "emula", poiché Turing è interessato a un'imitazione fedele – il comportamento intellettuale di un cervello umano può essa stessa essere appropriatamente descritta come un cervello, o come pensante" [Copeland, 2000]²¹⁷.

²¹⁶ "Our potential interest in the imitation game is aroused not by the fact that a computer might learn to play yet another game, but that in some way this test reveals a connection between possible computer activities and our ordinary concept of thinking"

²¹⁷ "Dub the following *Turing's Principle*: A machine that by means of calculation imitates – or, better, 'emulates,' for Turing is concerned with faithful imitation – the intellectual behaviour of a human brain can itself appropriately be described as a brain, or as thinking". Copeland nella sua analisi dell'*Imitation Game*, fa riferimento, oltre al "Principio di Turing" ad un'altra assunzione, che definisce il "*Principio di de Cordemoy*", dal nome del filosofo, storico, avvocato e seguace della filosofia cartesiana Gerard de Cordemoy (1926-1984). Questo stabilisce "Se tutti gli esperimenti che siamo in grado di fare mostrano che x usa il linguaggio come noi, allora x ha un'anima (cioè pensa)". ["If all the experiments that we are capable of making show that x uses speech as we do, then x has a soul (i.e. thinks), Copeland, 2000 trad. mia]. Il principio è tratto da un brano del pensatore cartesiano secentesco dal suo *A Philosophical Discourse Concerning Speech* del 1668 [de Cordemoy, 1688], traduzione postuma pubblicata a Londra dell'originale *Discours physique de laparole* del 1668. In esso de Cordemoy afferma: "Parlare non è ripetere le stesse parole che hanno colpito l'orecchio, ma pronunciarne altre con uno scopo e adatte a esse... [N]essuno dei corpi che producono echi pensa, anche se li sento ripetere le mie parole... Dovrei per la stessa ragione giudicare che i pappagalli non pensano... Ma per non esaminare ulteriormente come stanno le cose con i pappagalli e tanti altri corpi la cui forma è molto diversa dalla mia, continuerò l'indagine... [Riguardo a quelli] che mi assomigliano così perfettamente senza... Penso di poter... stabilire come principio che... se trovo attraverso tutti gli esperimenti che sono in grado di fare che usano il linguaggio come faccio io... ho una ragione infallibile per credere che abbiano

Turing, come abbiamo già notato, aveva come target del suo test le macchine non organizzate studiate in *Intelligent Machinery*, in particolare quelle di tipo P. In esse, oltre alla disciplina necessaria a farle diventare macchine universali, egli tentava di instillare quel residuo che, nello stesso saggio, definiva come la facoltà dell'“iniziativa”. Questo scarto – che Turing indicava come indispensabile per riprodurre nelle macchine l'intelligenza umana – è precisamente ciò che sfugge a una descrizione compiuta e formale, la quale sola permetterebbe di istanziarlo in una macchina di Turing.

un'anima come me." [" To speak is not to repeat the same words, which have struck the ear, but to utter others to their purpose and suitable to them. ... [N]one of the bodies that make echoes do think, though I hear them repeat my words ... I should by the same reason judge that parrets do not think neither. ... But not to examine any further, how it is with parrets, and so many other bodies, whose figure is very different from mine, I shall continue the inquiry ... [Concerning those] who resemble me so perfectly without ... I think I may ... establish for a Principle, that ... if I finde by all the experiments I am capable to make, that they use speech as I do, ... I have infallible reason to believe that they have a soul as I. [de Cordemoy, 1668, pp.13–14, in Copeland, 2000 trad. mia]" e "Capisco molto bene che un semplice meccanismo potrebbe pronunciare alcune parole, ma capisco allo stesso tempo che se gli organi che dovrebbero distribuire il vento, o aprire i tubi da cui dovrebbero uscire quelle voci, avessero un certo ordine stabilito tra di loro, non potrebbero mai cambiarlo, così che quando si sentisse la prima voce, quelle che di solito la seguono sarebbero necessariamente sentite anche loro, a condizione che il vento non manchi al meccanismo; mentre le parole che sento pronunciare da corpi fatti come il mio quasi non hanno mai lo stesso seguito." [" I do very well conceive, that a meer Engin might utter some words, yet I understand at the same time, that if the organs, which should distribute the wind, or open the pipes, whence those voices should issue, had a certain settled order among them, they could never change it, so that when the first voice were heard, those that were wont to follow it, would needs be heard also, provided the wind were not wanting to the Engin; whereas the words which I hear uttered by bodies made like mine have almost never the same sequel. De Cordemoy, 1688, p. 6 in Copeland, 2000 trad. mia]. La derivazione cartesiana di un test che è un concetto centrale nello sviluppo di scienze che hanno visto in Descartes se non un nemico giurato, ma almeno un obiettivo polemico privilegiato, può sembrare paradossale, ma sono diversi i luoghi nel pensiero cartesiano dove si possono trovare degli "antecedenti" che riportano alle riflessioni di Turing sull'*Imitation Game*. Nel *Discorso sul metodo* si può leggere: "Mi ero a questo punto soffermato in particolare per far vedere che, se vi fossero macchine tali da avere gli organi e la forma di una scimmia o di qualche altro animale privo di ragione, noi non avremmo alcun modo per riconoscere che esse non siano in tutto della stessa natura di questi animali. Se invece ve ne fossero che avessero un corpo simile al nostro e imitassero le nostre azioni quanto fosse moralmente possibile, noi avremmo sempre due mezzi assai certi per riconoscere che esse non sarebbero nonostante ciò dei veri uomini. Il primo di questi mezzi è che mai esse potrebbero usare parole, né altri segni componendoli come noi facciamo per manifestare agli altri i nostri pensieri. Si può infatti ben concepire che una macchina sia fatta in maniera tale da profferire delle parole, [...] Il secondo mezzo è che, per quanto esse possano fare molte cose tanto bene quanto ciascuno di noi, o anche meglio, esse fallirebbero inevitabilmente in altre, attraverso le quali si scoprirebbe che esse non agirebbero per conoscenza, ma solo in virtù della disposizione dei loro organi" [Cartesio, 1637, Parte V, p.57 trad. it.]. Il secondo luogo è nella *Meditatio seconda*, in cui il filosofo francese afferma: «Per questo concluderei subito: la cera è dunque conosciuta attraverso la visione dell'occhio, non attraverso lo sguardo della mente; se non fosse che ho appena volto lo sguardo, dalla finestra, su degli uomini che camminano per strada, ed anche essi dico di vederli, come abitualmente dico di vedere la cera. Che cosa vedo, però, oltre i cappelli e i vestiti sotto cui potrebbero nascondersi automi? Eppure, giudico che sono uomini. E così quel che ritenevo di vedere con gli occhi lo comprendo con la sola facoltà di giudicare, la quale è nella mia mente. E così quel che ritenevo di vedere con gli occhi lo comprendo con la sola facoltà di giudicare, la quale è nella mia mente" [Cartesio, 1641, *Meditazione seconda*, p.32 trad. it.]. Per un resoconto degli antecedenti nella storia della filosofia moderna del *test di Turing* si veda Bianchini, 2007. I riferimenti ai *pappagalli* danno ragione storica dell'impiego del termine nel dibattito contemporaneo sulla natura delle IA generative, degli LLMs e dei loro limiti.

Turing, parlando della possibilità che le macchine imparino (*Learning machines*), è netto a riguardo:

Un aspetto importante delle macchine che apprendono è che spesso il loro istruttore saprà assai poco di che cosa succede esattamente là dentro, anche se è possibile che, in una certa misura, sia ugualmente in grado di prevedere il comportamento del suo allievo. Questo si applica particolarmente all'istruzione ulteriore di una macchina che deriva da una macchina-bambina il cui progetto (o il cui programma) è sufficientemente collaudato. Ciò è in netto contrasto con quello che è il modo di procedere normale quando una macchina viene usata per fare calcoli: in quel caso, l'obiettivo è avere in mente un'immagine chiara dello stato della macchina in ciascun momento del calcolo. Tale obiettivo non può essere acquisito senza un bel po' di fatica. Da questo punto di vista, appare strana l'idea che «la macchina può fare soltanto ciò che sappiamo come ordinarle di eseguire». La gran parte dei programmi che possiamo mettere nella macchina faranno sì che essa faccia cose a cui non siamo in grado di dare un senso, o che consideriamo un comportamento del tutto casuale. Presumibilmente, il comportamento intelligente consiste in un allontanamento dal comportamento in tutto e per tutto disciplinato in cui consiste un calcolo; ma un allontanamento piuttosto piccolo, che non dà origine a un comportamento casuale né a circolarità [loops] ripetitive del tutto prive di senso [Turing, 1950, p.155 trad. it 1994 enfasi mie].

Le macchine-bambino [*child-machines*], le macchine non organizzate di tipo-P, sono quelle che rispondono all'esigenza di Turing di trovare un modello della struttura cerebrale dell'infante, più adatta per la modellizzazione nella macchina rispetto a quella dell'adulto: "Il cervello infantile è qualcosa di simile a un taccuino di quelli che si comprano dai cartolai: poco meccanismo e una quantità di fogli bianchi (meccanismo e scrittura sono dal nostro punto di vista quasi sinonimi)" [Turing, 1950, p.151 trad.it 1994].

La descrizione del tipo di macchine in questione iniziata in *Intelligent Machinery*, e che prosegue nei lavori seguenti di Turing sul tema, è, inoltre, un saggio sulle modalità di *machine learning* che conferma il modo in cui Turing ha anticipato il connessionismo moderno anche in questo ambito. [Copeland, 2004cc, p.402; Turing 1950 in Copleand, 2004, pp.465-467].

Turing trattando delle modalità con cui le macchine apprendono, "spiega come l'«intelligenza» non venga installata nel cervello, come un impianto elettrico già pronto" [Hodges, 1985, p.492 trad. it.]. La chiave dell'intelligenza risiede ne "la capacità di imparare. Solo una macchina capace di apprendere avrebbe segnato la svolta decisiva". [Hodges, 1985,, p.491 trad. it. enfasi originale].

Produrre direttamente una macchina universale non avrebbe, per Turing, potuto avere come risultato la riproduzione della capacità di “iniziativa” tipica invece del cervello umano. Le tavole di istruzione delle macchine di tipo P sono incomplete, la macchina deve “riuscire a farcela da sola» in un modo o nell’altro e cioè non dovesse acquistare la capacità di pensare grazie all’intervento di un essere superiore che gli scriveva i programmi” [Hodges, 1985,, p.491 trad. it.].

Come istruire la macchina a farcela da sola, nello stesso modo in cui ci riesce un cervello umano? Con un meccanismo di *piacere-dolore*, ricompense e premi che, come nota Hodges, il biografo di Turing, hanno una forte componente autobiografica nel percorso del matematico inglese. “I processi educativi che Alan ha in mente sono quelli di una *public school*: una pedagogia fondata sul sistema del bastone e della carota” [Hodges, 1985, p.491 trad. it.]. Turing aveva infatti studiato alla prestigiosa *Sherborne School*, una prestigiosa *public school* situata a Sherborne, nel Dorset, dal maggio 1926 al giugno 1931.

Nello stesso stesso modo in cui la corteccia non organizzata del neonato viene mediante un processo pedagogico, disciplinata in quella dell’adulto, così deve essere istruita la macchina

Copeland descrive e suggerisce che le idee, di ispirazione neurofisiologica di Turing di questo processo di apprendimento nel cervello siano antesignane del concetto oggi studiato dagli scienziati di «genesi sinaptica» [Copeland, 2012, p. 271, trad. it.].

Gli ammassi di cellule/macchine sintetiche non connesse tra loro, o connesse in modo casuale, si addestravano, e formavano cioè nuove connessioni, «attraverso la rottura di connessioni intercellulari già esistenti» [Copeland, 2012, p. 271, trad. it.].

Questo processo, la “potatura sinaptica” [Copeland, 2012, p. 271, trad. it.], veniva considerato da Turing un meccanismo essenziale per lo sviluppo dell’apprendimento, in quanto permetteva alla rete neurale sintetica di selezionare le connessioni più utili e di eliminare quelle inefficienti. In questo modo, un sistema inizialmente disorganizzato poteva progressivamente affinarsi e specializzarsi, acquisendo la capacità di eseguire operazioni complesse. Tale dinamica è alla base delle moderne reti neurali artificiali, che attraverso l’ottimizzazione delle connessioni riescono a migliorare la loro performance computazionale. “Attraverso questi processi gemelli di creazione e rottura di connessioni, il groviglio casuale di cellule nervose artificiali può essere affinato fino a diventare una macchina per portare a termine specifici compiti intellettuali” [Copeland, 2012, p. 271, trad. it.].

Se il *Machine Learning* – che come abbiamo visto era un tema chiave già all’interno della cibernetica – è definibile come quel processo di apprendimento che “si occupa di costruire

sistemi che migliorano le loro prestazioni in un compito quando vengono forniti esempi di prestazione ideale per quel compito, oppure che migliorano le loro prestazioni attraverso l'esperienza ripetuta nello svolgimento del compito" [Bringsjord, Govindarajulu, 2018]²¹⁸, le macchine non organizzate di Turing di tipo A, B e P si pongono pienamente in questo solco.

Per Turing "Era infatti lo studio degli «stati mentali» che [...] avrebbe condotto a "costruire un cervello»"²¹⁹ [Hodges, 1985, p.492 trad. it.]. E per fare ciò aveva bisogno di un criterio di valutazione. Quest'idea, *insegnare* a una macchina a migliorare il suo comportamento fino a trasformarlo in "intelligenza", è la chiave per capire quasi tutte le proposte positive che vengono avanzate nel rapporto. Questa volta infatti Alan intendeva fare un uso costruttivo del principio dell'imitazione" [Hodges, 1985, p.492 trad. it.].

L'*Imitation Game* sarebbe allora un modo per valutare l'intelligenza della macchina che rispetti l'assunto– come Turing affermerà in una conferenza del 1947 alla *London Mathematical*

²¹⁸ "Machine learning is concerned with building systems that improve their performance on a task when given examples of ideal performance on the task, or improve their performance with repeated experience on the task".

²¹⁹ L'interesse di Turing per la costruzione di un cervello, o di un suo equivalente meccanico, era condiviso con Claude Shannon ed era uno dei pochi argomenti di cui i due uomini, impegnati nei progetti crittografici nella lotta contro il nemico nazista, potevano parlare liberamente. Racconta Hodges che, nel 1943, durante un pranzo, Turing esclamò ironicamente: «Shannon non vuole dare solo dei dati al suo Cervello, vuole dargli cultura! Vuole offrirgli della musica!». In un'altra occasione, Shannon tenne un discorso sulla «macchina pensante», affermando provocatoriamente: «Non mi interessa fare un cervello potente, voglio solo ottenere un cervello mediocre, diciamo come quello del presidente dell'American Telephone and Telegraph Company». La battuta lasciò la sala in un silenzio imbarazzato, mentre Shannon proseguiva a spiegare la sua idea: una macchina alimentata con dati su prezzi industriali e mercati finanziari, che avrebbe risposto alla domanda «Devo vendere o comprare?». [Hodges, 1985, pp.328-329 trad.it.]. Questo tema ci conduce a una questione controversa: quanto spazio Turing abbia effettivamente riconosciuto alla dimensione dell'*embodiment*, a fronte della sua insistenza sull'intelligenza come organizzazione funzionale indipendente dal sostrato che la implementa. Si tratta infatti di stabilire quanto sia rilevante, per il progetto di "far fare esperienza" alle macchine – intesa come processo di modificazione finalizzato all'organizzazione – il ruolo del corpo e dell'interazione col mondo, al fine di un addestramento capace di condurle a un'intelligenza paragonabile a quella che le facoltà del cervello consentono agli esseri umani. La questione è, a mio avviso, aperta, in quanto in *Intelligent Machinery*, Turing ipotizza, parlando della possibilità di costruire macchine pensanti, una sorta di essere cibernetico in cui le parti umane sono state sostituite da parti meccaniche. "Un modo di affrontare l'impresa di costruire una «macchina pensante» sarebbe quello di prendere una persona intera e cercare di rimpiazzare tutte le sue parti con sostituti meccanici: telecamere, microfoni, altoparlanti, ruote e servomeccanismi di gestione, come anche un qualche tipo di «cervello elettronico»." [Turing, 1948, pp. 103 trad. it.]. L'essere in questione, di proporzioni, allo stato dell'arte della tecnologia nel 1948, sarebbe stato "dimensioni immense, anche se la parte del «cervello» fosse fissa e controllasse il corpo a distanza" [Turing, 1948, pp. 103 trad. it.]. Qui, oltre a notare la somiglianza con i sistemi di *cloud computing* contemporanei, dove l'elaborazione complessa è centralizzata rispetto ai vari meccanismi locali, è da sottolineare come Turing ponga l'accento sulla necessità che questo essere per "poter avere la possibilità di farsi un'esperienza, dovrebbe essergli permesso di aggirarsi per la campagna con serio pericolo per gli abitanti «normali». Inoltre, anche se le fossero fornite tutte le caratteristiche che viste prima, la creatura ancora non avrebbe esperienza di cibo, sesso, sport e molte altre cose di interesse per gli esseri umani"[Turing, 1948, pp. 103 trad. it.]. Lo stesso paragone tra una macchina pensante e il cervello umano mostra i suoi limiti: come osserva Turing, non sarebbe realistico pretendere che una macchina appena costruita possa competere alla pari con un essere umano già formato dall'esperienza. A questo proposito egli scrive: "non sarebbe leale aspettarsi che una macchina uscita fresca dalla fabbrica competi con un laureato su un piede di parità. Il laureato ha avuto contatti con altri esseri umani per venti e più anni, contatto che ha continuato a modificare il suo modello di comportamento lungo tutta la vita" [Turing, 1948, pp. 104-105, trad. it.].

Society avente per tema il suo lavoro sull'ACE – per cui “la macchina deve essere trattata in modo equo e leale” [Turing, 1947, p.86 trad. it.]. “Se ci si aspetta che la macchina sia infallibile, allora essa non può essere anche intelligente” [Turing, 1947, p.86 trad. it.]. Non passerebbe il Test, in quanto nessun giudice la scambierebbe per un essere umano, e questo non per qualche strategia di dissimulazione.

Ma valutare con *fair play* le macchine significa avere per riferimento il modello di intelligenza rappresentato dal cervello umano, che la macchina vorrebbe descrivere. “Anche il matematico prende qualche cantonata quando sperimenta nuove tecniche” [Turing, 1947, p.86 trad. it.]. La macchina per esibire intelligenza non può essere infallibile, perché non è infallibile il modello che tenta di emulare:

Per proseguire nel mio appello per un *fair play* nei confronti delle macchine nel verificare il loro quoziente di intelligenza, voglio osservare che ogni matematico umano è sempre sottoposto a un addestramento prolungato, e che l'effetto dell'addestramento può essere considerato non dissimile dall'inserzione di tavole di istruzioni dentro a una macchina [Turing, 1947, p.87 trad. it.].

Per questo motivo nella versione embrionale del test del 1948 Turing parlava di “giocatori mediocri di scacchi”.

Il Test di Turing, acquista il suo valore, allora, nell'essere come una “potenziale fonte di buona evidenza induttiva per l'ipotesi che le macchine pensino” [Moor, 1976 trad. mia]²²⁰. Ma di quali macchine si tratta? Di quelle che, organizzandosi come macchine universali, cercano di costituirsi come descrizioni del cervello e, in questo senso, “pensano”: vale a dire, mirano ad acquisire la facoltà dell'iniziativa e ad agire in un modo – nello schema esemplificato dalle domande e risposte dell'*Imitation Game* – che può essere considerato intelligente.

Questa idea, il suo percorso genetico²²¹ gli sviluppi che ne sono derivati hanno avuto conseguenze di enorme rilievo nello sviluppo delle scienze cognitive, nei diversi paradigmi

²²⁰ “The real value of the imitation game lies not in treating it as the basis for an operational definition but in considering it as a potential source of good inductive evidence for the hypothesis that machines think”. A questo proposito si veda anche Moor, 2001.

²²¹ Turing, in *Intelligent Machinery*, primo “manifesto dell'Intelligenza Artificiale”, non solo offre una versione ristretta dell'*Imitation Game*, ma anticipa anche temi portanti del paradigma connessionista. Introduce l'idea che “l'attività intellettuale consiste principalmente in vari tipi di ricerca”, concetto che sarà centrale nei modelli di problem-solving di Newell e Simon. Inoltre, ipotizza un metodo di ottimizzazione basato sulla selezione naturale, anticipando gli algoritmi genetici. Infine, delinea un approccio alla risoluzione dei problemi basato sulla logica formale, che diventerà il fondamento delle prime architetture dell'IA simbolica. Copeland, 2004, p.401.

dell'intelligenza artificiale e nell'evoluzione della computer science, tanto sul versante del software quanto su quello dell'hardware.

La possibilità per una macchina di essere considerata intelligente dipende, come abbiamo visto, anche dalla capacità di esibire comportamenti che la qualificano come tale ma che esulano totalmente dalle possibilità di una macchina di Turing: l'elasticità, la possibilità di commettere errori, la gestione di input ambigui o la tolleranza ai danni [Copeland, 2012, pp. 272-273, trad. it.]. Tutto ciò non è realizzabile per una MT "disciplinata", nella misura in cui essa incarna le caratteristiche di una procedura algoritmica rigorosa. Da questo punto di vista, Turing si pone come precursore del paradigma connessionista come sopra descritto.

Ma c'è di più. Le macchine che descrivono il cervello, in tutte le facoltà necessarie all'intelligenza, esibiscono una forma di pensiero. Cos'è per Turing il pensiero? Non è che un "ronzio" [*buzzing*]. "Non voglio dare una definizione del pensiero, ma se dovessi farlo probabilmente non saprei dire nulla di più se non che è una sorta di ronzio che avviene dentro la mia testa" [Turing 1952, in Copeland, 2004c, p.494 trad. mia]²²².

A causa dell'impossibilità, la stessa riscontrata due anni prima in *Computing Machinery and Intelligence*, di specificare con chiarezza e senza ambiguità cosa sia il pensiero, è possibile tralasciarne una definizione specifica.

La cosa importante è cercare di tracciare una linea tra le proprietà di un cervello, o di un uomo, che vogliamo discutere, e quelle che non ci interessano. Per fare un esempio estremo, non siamo interessati al fatto che il cervello abbia la consistenza del porridge freddo. Non vogliamo dire 'Questa macchina è piuttosto dura, quindi non è un cervello, e quindi non può pensare' [Turing 1952, in Copeland, 2004c, pp.494-495 trad. mia]²²³

Il pensiero, così come può essere testato, non bada alle differenze tra pappa d'avena, tubi termoionici, neuroproteina o silicio. Ciò che conta sono le proprietà descrivibili. In ciò Turing si propone come il capostipite del *Funzionalismo*²²⁴, cioè, come lo abbiamo definito nelle pagine

²²² "I don't want to give a definition of thinking, but if I had to I should probably be unable to say anything more about it than that it was a sort of buzzing that went on inside my head".

²²³ "I don't want to give a definition of thinking, but if I had to I should probably be unable to say anything more about it than that it was a sort of buzzing that went on inside my head. But I don't really see that we need to agree on a definition at all. The important thing is to try to draw a line between the properties of a brain, or of a man, that we want to discuss, and those that we don't. To take an extreme case, we are not interested in the fact that the brain has the consistency of cold porridge. We don't want to say 'This machine's quite hard, so it isn't a brain, and so it can't think.'"

²²⁴ Sul concetto di Funzionalismo si vedano Block, 1980; Piccinini 2010 e 2024 in cui si analizza il rapporto fra F. e teorie computazionali della mente criticandone l'equivalenza e Di Francesco, 2003, pp. 127-173. Una posizione

precedentei quella cioè quella corrente di filosofia della mente e delle scienze cognitive che ritengono che “i criteri per l’identificazione degli stati mentali vadano individuati in base alle relazioni funzionali degli stati stessi” [Frixione, 1992, p. 18], indipendentemente da cosa li implementi. È la funzione che fa uno stato mentale, non la costituzione interna [Levin, 2023].

La macchina di Turing – alle opportune condizioni, nella misura in cui è descrizione del cervello – rappresenterebbe la struttura del pensiero. In questa prospettiva, il funzionalismo è quell’approccio per cui “da un lato la mente è immateriale nel senso in cui lo è una struttura formale e il pensiero è una sorta di calcolo logico, ma dall’altro la materia può pensare, a patto che essa abbia una particolare organizzazione funzionale” [Di Francesco, 2003, p. 26]. Tale condizione di indifferenza delle relazioni funzionali al “sostrato materiale che le realizza” [Di Francesco, 2003, p. 149] rispetta la natura astratta delle macchine di Turing.

Secondo quanto abbiamo indicato sopra come *Principio di Turing*, a una macchina capace di descrivere mediante calcolo il cervello deve essere ascrivito il pensiero. Quest’ultimo risulterebbe, allora, una manipolazione computazionale, e ciò che viene computato sono simboli: la macchina di Turing ne fornisce un modello astratto.

La macchina di Turing, come sappiamo, è dotata di un sistema di lettura/scrittura, la testina “Δ”, che rappresenta “un rudimentale occhio-mano” [Di Francesco, 2003, p.139]. Quest’occhio-mano, la scarnificazione di un essere umano che compie un’operazione di calcolo, può eseguire solo le operazioni della quintupla. «L’occhio-mano sopra descritto è in realtà un omuncolo un po’ stupido [...] ma l’aspetto importante per noi è che esso può portare a termine una dimostrazione sulla semplice base delle proprietà della forma dei simboli che incontra» [Di Francesco, 2003, p.142].

L’Intelligenza Artificiale Classica, la GOF AI, (*Good-Old-Fashioned-Artificial-Intelligence*) si costituisce proprio intorno a due assunti, e cioè che: “1) la nostra capacità di trattare le cose in modo intelligente è dovuta alla nostra capacità di pensare ad esse in modo ragionevole (compreso in questa accezione il pensiero subconscio); e 2) la nostra capacità di pensare alle cose in modo ragionevole corrisponde a una facoltà di manipolazione “automatica” interna di simboli” [Haugeland, 1988, p.88 trad. it.]. L’intelligenza allora “dipende solo dall’organizzazione di un sistema e dal suo operato come manipolatore di simboli” [Haugeland, 1988, p.12]²²⁵.

“classica” di Funzionalismo legato alla macchina di Turing prima che ne diventasse un critico severo, è quella di Putnam, 1975 in cui si sostiene che la mente è una macchina di Turing e si identificano gli stati mentali con le quintuple della MT.

²²⁵ L’introduzione dell’“omuncolo”, di un manipolatore, più o meno intelligente, di simboli fa incorrere questa prospettiva nel “paradosso della ragione meccanica” [Haugeland, 1985, p. 39, trad. it.]. Questo problema si pone

A questa visione dell'intelligenza e della sua riproduzione nella macchina si ispira anche quello che può essere considerato l'atto di nascita dell'*Intelligenza Artificiale*, o almeno la prima occasione in cui il termine compare ufficialmente²²⁶: la proposta di ricerca avanzata da un

nel momento in cui si cerca di dar conto del pensiero come manipolazione di simboli, astruendo totalmente dalla dimensione semantica. Si implica, cioè, la presenza di un *homunculus*, un'entità interna che "interpreta" i simboli. Ma questo porta a un regresso all'infinito: per spiegare il pensiero, si postula un altro pensatore all'interno del sistema, che a sua volta richiederebbe un altro interprete, e così via all'infinito. Haugeland propone una soluzione a questo paradosso, e cioè l'ipotesi per cui "gli *homuncoli* diventino ad ogni stadio sempre meno brillanti, fino a giungere a un punto a cui non hanno più intelligenza" [Haugeland, 1985, p. 112 trad. it.]. Questa ipotesi richiama da vicino il paradosso turinghiano delle "sfoglie di cipolla" [Cordeschi, Tamburrini, 2015]. L'idea è quella di un progressivo avvicinarsi a uno strato meccanico in realtà inattuabile: "Anche l'analogia della «sfoglia della cipolla» può servire. Considerando le funzioni mentali o cerebrali troviamo certe operazioni che possiamo spiegare in termini puramente meccanici. Questo, diciamo, non corrisponde alla mente come essa è in realtà: è una specie di sfoglia che dobbiamo togliere se vogliamo trovare la mente reale. Ma poi in ciò che rimane troviamo un'altra sfoglia da togliere, e così via. Procedendo in questo modo arriviamo infine alla mente «reale», o solo a una sfoglia che non contiene nulla?" [Turing, 1950, pp. 149-150, trad. it.]. Questo paradosso porta al centro critico del paradosso della ragione meccanica, ossia il "mistero del significato originale": da dove traggono il loro significato originale i simboli? Questo problema ha avuto la sua formulazione classica in Harnad [Harnad, 1990], come "*The Symbol Grounding Problem*", e cioè il problema su "come si può rendere l'interpretazione semantica di un sistema formale di simboli intrinseca al sistema stesso, invece di essere semplicemente parassitaria rispetto ai significati nelle nostre menti? Come possono i significati dei simboli privi di senso, manipolati esclusivamente in base alle loro forme (arbitrarie), essere ancorati a qualcosa che non sia solo altri simboli privi di senso?" [Harnad, 1990, trad. mia]. La proposta di risoluzione di Harnad richiama l'idea di *radicare (grounding)* nel mondo i simboli attraverso rappresentazioni non simboliche: *iconiche* (analoghi sensoriali diretti) e *categoriali* (astrazioni degli invarianti percettivi). È interessante notare come Haugeland, in modo indiretto, e Harnad, più esplicitamente, proponano soluzioni che implicano un superamento della struttura computazionale classica/seriale: il primo ipotizzando un sistema basato su livelli di agenti sempre più semplici e decentralizzati, il secondo valorizzando le strutture connessioniste, in un modello di compromesso col paradigma simbolico, che risulta particolarmente adatto a identificare invarianti percettivi, riconoscere strutture ricorrenti nei dati sensoriali e categorizzarli. Floridi [Floridi2005] pone questo problema tra i principali che la *Filosofia dell'Informazione* deve affrontare nel suo sviluppo. Luciano Floridi considera il problema della fondazione dei simboli (PFS) una delle questioni aperte più rilevanti nella filosofia dell'informazione. Il PFS riguarda il modo in cui i dati possano acquisire significato senza presupporre capacità semantiche innate (*innatismo*) o un interprete esterno (*esternalismo*), evitando così di eludere la questione-un vincolo che Floridi formalizza nella condizione di impegno semantico nullo (N). Tra le possibili strategie, egli esplora un approccio ibrido, che integra interazione sensomotoria (*bottom-up*) con processi di astrazione e comunicazione (*top-down*). Pur non offrendo una soluzione definitiva, questa prospettiva suggerisce che la semantica possa emergere gradualmente attraverso meccanismi evolutivi e senza presupporre significati già presenti [Floridi, 2010, pp. 134-161].

²²⁶ Per la storia della Conferenza di Dartmouth si rimanda a McCorduck, 2004, pp. 11-136. John McCarthy – che può essere indicato come l'inventore ufficiale del nome – insistette sull'uso del termine *intelligenza artificiale* per distinguerlo dalla *teoria degli automi* (dal titolo di una raccolta di saggi da lui curata insieme a Shannon), che rappresentava un ambito più tecnico della teoria matematica. La scelta del nome rifletteva la sua fascinazione per l'idea secondo cui vi "era possibilità di esprimere qualsiasi problema intellettuale in termini di una macchina di Turing, il modello ipotetico in cui una stringa di simboli viene processata per generare una nuova stringa di simboli [...] [e] invertire questo processo: osservare la stringa di simboli in uscita per capire cosa fosse stato dato in ingresso" ["And he remembers being particularly taken with an idea of McCarthy's that summer, of the possibilities of expressing any intellectual problems in terms of a Turing machine, that hypothetical machine where a string of symbols goes in to be processed and later a string of other symbols comes out. McCarthy suggested that this process be inverted, that to look at the string of symbols that came out might give you an idea of what had gone in." in McCorduck, 2004, p. 116, trad. mia]. Molti partecipanti, tuttavia, non gradivano allora – e non gradiscono tutt'ora – tale espressione. Arthur Samuel, che nel 1959 introdusse il termine *machine learning* per il suo programma di dama sviluppato all'IBM, riteneva che l'aggettivo *artificiale* suggerisse qualcosa di falso; mentre Newell e Simon preferivano parlare di *elaborazione complessa delle informazioni*. Nonostante tali resistenze, la Conferenza di Dartmouth consacrò definitivamente il termine *intelligenza artificiale*, rendendolo il riferimento dominante nel campo.

gruppo di giovani scienziati-John McCarthy (1927-2011), Marvin Minsky (1927-2016), Nathaniel Rochester (1919-2001) e Claude Shannon - destinati a diventare tra le figure più prestigiose del campo.

Il documento, *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*, [McCarthy et al., 1956] consisteva in una richiesta di finanziamento di 13.500 dollari alla Rockefeller Foundation per una sessione di studi da tenersi nell'estate del 1956 presso il Dartmouth College, con l'obiettivo di discutere la possibilità di sviluppare programmi per computer in grado di *comportarsi o pensare* in modo intelligente.

Infatti, nella proposta possiamo leggere:

Proponiamo che durante l'estate, al Dartmouth College di Hanover, New Hampshire, si svolga uno studio della durata di un mese-uomo sull'intelligenza artificiale. Lo studio dovrà procedere sulla base della congettura che ogni aspetto dell'apprendimento, o qualunque altra caratteristica dell'intelligenza, possa in linea di principio essere descritto in modo così preciso da consentire la costruzione di una macchina in grado di simularlo. Si tenterà di scoprire come fare in modo che le macchine utilizzino il linguaggio, formino astrazioni e concetti, risolvano tipi di problemi oggi riservati agli esseri umani e migliorino sé stesse [McCarthy et al., 1956 trad. mia]²²⁷.

L'intelligenza, allora, è un fenomeno formalizzabile e, pertanto, riproducibile attraverso un sistema artificiale, indipendentemente dalla struttura fisica che la realizza.

Malgrado le differenze di opinione tra i promotori²²⁸-come quelle tra chi riteneva che l'IA dovesse basarsi esclusivamente sulla logica matematica senza preoccuparsi se questo fosse il modo in cui ragiona la mente umana (McCarthy) e chi invece sosteneva che le macchine dovessero imitare il funzionamento della mente umana, ritenuto diverso dalla logica matematica (Minsky) [Frixione, 1992, p21] -il punto fondamentale resta invariato. Il pensiero, nelle sue manifestazioni simboliche di alto livello, e in ogni altra sua forma, è indipendente dal substrato biologico ed è, in linea di principio, riproducibile da una macchina.

²²⁷ "We propose that a month man study of arti cial intelligence be carried out during the summer of at Dartmouth College in Hanover New Hampshire The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it An attempt will be made to nd how to make machines use language form abstractions and concepts solve kinds of problems now reserved for humans and improve themselves".

²²⁸ Una efficace sintesi delle divergenze teoriche dei promotori dell'evento di Dartmouth si trova in Gardner, 1985, pp.168-178 trad. it..

Da questo punto di vista l'operazione computazionale della Macchina di Turing sui simboli, il pensiero, può essere interpretata alla luce di "una versione "estremista", o "radicale" della Tesi di Church, che, grosso modo, potrebbe essere formulata come segue: ogni attività linguistico-cognitiva è calcolabile da una macchina di Turing" [Frixione, 1992, p.16, enfasi originale].

L'Imitation Game sarebbe allora una modalità per mettere alla prova questa versione «estremista» della Tesi di Church-Turing. Si chiama in causa qui un'istanza che va oltre i limiti di questa tesi sulla computazione, "poiché afferma specificamente che la macchina intelligente è un sistema simbolico, facendo così un'affermazione architetturale specifica sulla natura dei sistemi intelligenti" [Newell, Simon, 1976, p.88 trad. mia]²²⁹.

Prendiamo in considerazione un sistema che contiene una raccolta di simboli, "schemi fisici che possono verificarsi come componenti di un altro tipo di entità chiamato espressione (o struttura simbolica)" [Newell, Simon, 1976, p.85 trad. mia]²³⁰. I simboli sono legati in strutture in modo fisico, intendendo con questo che: "(1) obbediscono chiaramente alle leggi della fisica e sono realizzabili come sistemi ingegnerizzati fatti di componenti progettati; (2) sebbene il termine "simbolo" prefiguri una determinata interpretazione, non è limitato ai soli sistemi simbolici umani" [Newell, Simon, 1976, p.85 trad. mia]²³¹

Inoltre, il sistema simbolico così strutturato "include anche una serie di processi che operano su di esse per generare nuove espressioni: processi di creazione, modifica, riproduzione e distruzione.". Il sistema simbolico fisico, la struttura di manipolazione di simboli, "può essere visto come una macchina che, nel tempo, produce e modifica una collezione evolutiva di strutture simboliche" [Newell, Simon, 1976, pp.85-86 trad. mia]²³².

²²⁹ "For it asserts specifically that the intelligent machine is a symbol system, thus making a specific architectural assertion about the nature of intelligent systems".

²³⁰ "A physical symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure)".

²³¹ "Let us return to the topic of symbols, and define a physical symbol system. The adjective "physical" denotes two important features: (1) such systems clearly obey the laws of physics-they are realizable by engineered systems made of engineered components; and (2) although our use of the term "symbol" prefigures our intended interpretation, it is not restricted to human symbol systems"

²³² "A physical symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure). Thus a symbol structure is composed of a number of instances (or tokens) of symbols related in some physical way (such as one token being next to another). At any instant of time the system will contain a collection of these symbol structures. Besides these structures, the system also contains a collection of processes that operate on expressions to produce other expressions: processes of creation, modification, reproduction, and destruction. A physical symbol system is a machine that produces through time an evolving collection of symbol structures".

Questa macchina di manipolazione simbolica altro non è che la Macchina di Turing. La quale appartiene alla categoria delle macchine “discrete” (un concetto che analizzeremo nel capitolo successivo) cioè quelle per cui “i suoi stati si prestano a essere descritti in modo naturale come un insieme discreto, in modo che il movimento della macchina consista nel saltare da uno stato all’altro”, e delle macchine di “controllo”, se “manipola solo informazione” [Turing, 1948, pp. 91-92, trad. it.].

L’operazione su un nastro infinito attraverso un sistema di lettura e scrittura regolato da una funzione di transizione (la configurazione, l’astrazione dello stato mentale dell’essere umano che computa) esemplifica il tipo di sistema simbolico fisico descritto. “Un sistema simbolico fisico è un’istanza di una macchina universale. Dunque, l’ipotesi del sistema simbolico implica che l’intelligenza sarà realizzata da un computer universale.” [Newell, Simon, 1976, p.88 trad. mia]²³³

Dal momento che da questo punto di vista la manipolazione computazionale di simboli è il pensiero, “un sistema simbolico fisico è in possesso degli strumenti necessari e sufficienti per l’azione intelligente generale” [Newell, Simon, 1976, p.89 trad. mia]²³⁴.

Ciò che conta, ai fini della realizzazione e della descrizione del pensiero, indipendentemente dal sostrato fisico che lo implementa, è la *manipolazione computazionale* di simboli. A questo livello l’*Imitation Game* testerebbe la porzione simbolica che si esprime nel linguaggio.

Come sistema manipolatore di simboli, un sistema fisico, “un meccanismo che elabora informazioni [...] è un meccanismo che elabora *rappresentazioni*, talvolta dette *simboliche*, assimilabili a “immagini interne” (dell’ambiente e del sistema stesso), di cui non necessariamente il sistema è consapevole, custodite in qualche senso del termine nella *memoria* del sistema” [Datteri, 2017, p.59].

Le rappresentazioni, in linea con le strutture di retroazione cibernetica analizzate nel primo capitolo, svolgono inoltre un ruolo adattivo fondamentale. Come afferma il cibernetico e

²³³ A physical symbol system is an instance of a universal machine. Thus the symbol-system hypothesis implies that intelligence will be realized by a universal computer

²³⁴ “By ‘necessary’ we mean that any system that exhibits general intelligence will prove upon analysis to be a physical symbol system. By ‘sufficient’ we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence. By ‘general intelligent action’ we wish to indicate the same scope of intelligence as we see in human action: that in any real situation, behavior appropriate to the ends of the system and adaptive to the demands of the environment can occur, within some limits of speed and complexity”. A proposito della *Physical-System-Symbol-Hypothesis* si vedano Di Francesco, 2003, pp.128-129; Clark, 1989, pp.11-13.

psicologo scozzese Kenneth Craik [1914-1945] nel suo testo *The Nature of Explanation* [Craik, 1943]:

Assumendo dunque l'esistenza del mondo esterno, ho delineato una teoria simbolica del pensiero, nella quale il sistema nervoso è visto come una macchina di calcolo capace di modellare o riprodurre in parallelo [*paralleling*] gli eventi esterni, e ho suggerito che questo processo di parallelismo sia la caratteristica fondamentale del pensiero e della spiegazione. Il possessore di un sistema nervoso è quindi in grado di anticipare gli eventi invece di affidarsi esclusivamente alla sperimentazione empirica invariabile [Craik, 1943, pp.120-121 trad. mia]²³⁵.

Vi sarebbe allora un livello intermedio, elaborato dall'equivalente di una macchina computazionale istanziata dal sistema nervoso, tra lo strato neurofisiologico – o qualsiasi altro equivalente, data l'indifferenza del sostrato – e quello del comportamento.

La natura adattiva della struttura simbolica del pensiero, che si esprime nel comportamento, potrebbe dunque rappresentare quell'iniziativa che l'*Imitation Game* vorrebbe mettere alla prova.

In realtà, continuando a considerare il gioco dell'imitazione come criterio di analisi delle caratteristiche delle teorie qui analizzate, possiamo considerare il Test di Turing in un'altra prospettiva. Più che un test sull'intelligenza delle macchine – qui abbiamo messo in evidenza come esso rappresenta una prova induttiva che coinvolge la nostra teoria dell'attribuzione del pensiero - “la prova elaborata dal matematico inglese apre la strada a una lettura nella direzione opposta: essa non ci dice solo come fa una macchina a pensare, ma suggerisce una possibile risposta al quesito su *come facciamo noi a pensare*” [Di Francesco, 2003, p.142, enfasi originale].

Questa prospettiva spiegherebbe le performance degli LLM odierni nel battere il Test di Turing [Biever, 2023]. Le reti neurali, che analizzeremo meglio più avanti, dei sistemi di *Deep Learning*, ispirate alla struttura fisiologica del cervello, sembrano essere più aderenti alle caratteristiche - alle *facoltà* - che Turing voleva emulare con le sue macchine e testare attraverso l'*Imitation Game*. Caratteristiche e facoltà che, come abbiamo visto, pongono il matematico britannico tra i pionieri del connessionismo.

²³⁵ “Assuming then the existence of the external world I have outlined a symbolic theory of thought, in which the nervous system is viewed as a calculating machine capable of modelling or paralleling external events, and have suggested that this process of paralleling is the basic feature of thought and of explanation. The possessor of a nervous system is thus able to anticipate events instead of making invariable empirical trial”

Gli LLM²³⁶, allora, supererebbero il test non per semplice avanzamento tecnologico, ma perché possiedono alcune delle qualità che Turing voleva mettere alla prova: la loro plasticità computazionale, la capacità di gestire input sotto-determinati e strutturalmente ambigui, e il fatto che i loro output probabilistici sembrano avvicinarsi a ciò che, nella nostra teoria del pensiero, ci aspettiamo dall'archetipo di entità a cui attribuiamo l'intelligenza, l'essere umano.

5. Scacco matto: Un matematico tra simboli, neuroni e calcolatori

La proposta di Turing, che emerge dall'analisi delle diverse tipologie di macchine computazionali ed è messa alla prova attraverso l'*Imitation Game*, si configura come una possibile risposta a uno dei problemi centrali delle scienze cognitive e della filosofia della mente: la conciliazione tra la prospettiva *top-down*, propria del funzionalismo e dell'Intelligenza Artificiale simbolica, secondo cui la cognizione dipende da processi sequenziali su simboli (come il linguaggio), e la prospettiva *bottom-up*, che la considera come pattern di attivazione neuronale [Frixione, 1992, p. 47; Copeland, 2004, p. 407].

Nel corso delle sue ricerche, Turing avanzò l'ipotesi – già delineata in *Intelligent Machinery* – che il cervello potesse essere, almeno in parte, una macchina di calcolo universale, implementata in una rete neurale e dotata di una specifica capacità di memoria. In questa prospettiva, la corteccia cerebrale, proprio in quanto rete neurale funzionante come una macchina universale, sarebbe in grado di sostenere l'elaborazione sequenziale e simbolica tipica dell'approccio top-down.

Ciò spiegherebbe come un sistema costituito da attivazioni neurali distribuite possa dar luogo a processi computazionali strutturati, capaci di manipolare simboli e di generare le facoltà cognitive che associamo all'intelligenza umana [Copeland, 1999; 2004, p. 407].

²³⁶ Sul rapporto tra i Large Language Models (LLM) -una storia che può essere fatta risalire a chatbot come ELIZA di Weizenbaum [Weizenbaum, 1966] e PARRY [Colby et al., 1972] o SHRDLU di Terry Winograd [Winograd, 1970 e 1972] estendibile sino all'AI generativa attuale, si vedano Marconi, 2025 e Floridi, Chiriatti, 2020. In quest'ultimo lavoro si analizza come il Test di Turing fornisca condizioni necessarie ma non sufficienti per l'intelligenza: non superarlo esclude un'IA dall'essere intelligente, ma superarlo non è sufficiente a qualificarla come tale. Un'analisi dei chatbot sopra indicati, e delle loro caratteristiche, rispetto alle conseguenze del pensiero di Turing sul tema dell'IA si veda Copeland, 2012, pp. 233-248. Su SHRDLU si vedano anche Haugeland, 1985, pp.174-172 trad. it e Boden, 2006, pp.685-697. Quest'ultimo programma, progettato per la comunicazione in lingua naturale (l'inglese) è emblematico del concetto di *micromondo*: "un dominio artificiale limitato, in cui i possibili oggetti, e possibili proprietà e i possibili eventi sono definiti in anticipo in modo ristretto ed esplicito" [Haugeland, 1985, p.173 trad. it.].

Nel caso della cognizione umana la descrizione del cervello sarebbe il *software*, e la corteccia cerebrale, il sostrato in cui è implementata, l'*hardware*. Se le macchine di Turing sono macchine di manipolazione dell'informazione, per il matematico inglese "i cervelli cadono quasi esattamente in questa categoria" [Turing, 1948, p. 92 trad. it.]. Ecco perché il lavoro del matematico inglese "rende così concepibile l'ambizione, tipica dell'intelligenza artificiale, di progettare delle macchine intelligenti (pensanti)" [Di Francesco, 2003, p. 26].

Possiamo comprendere allora, indipendentemente dall'angolo di analisi adottato rispetto al suo lavoro, come Turing abbia dato il via al "*programma di ricerca*", dal punto di vista "modellistico, epistemologico, metodologico e anche tecnologico," dell'Intelligenza Artificiale [Cordeschi, Tamburrini, 2015]. Programma di ricerca che si sviluppa intorno a 5 tesi:

- 1) *Funzionalismo*: L'intelligenza e l'adattatività dipendono dall'organizzazione funzionale, non dalla struttura biologica, e possono manifestarsi in macchine adeguatamente progettate.
- 2) *Metodo sintetico*: Costruire artefatti con organizzazione funzionale simile a quella biologica permette di testare empiricamente le ipotesi sull'intelligenza, fornendo prove di sufficienza (debole) o modelli esplicativi (forte).
- 3) *Uniformità del metodo*: Lo studio sperimentale può estendersi a processi mentali complessi come la creatività, l'intuizione e la coscienza.
- 4) *Rappresentazionalismo*: L'elaborazione simbolica e la rappresentazione del mondo sono essenziali per il comportamento intelligente.
- 5) *Agentività*: Le macchine possono essere analizzate secondo principi di razionalità e scopo adottati da un osservatore come avviene per gli esseri viventi umani o non umani. [Cordeschi, Tamburrini, 2015]

Ritroviamo alcuni di questi punti al cuore dell'impianto metodologico della scienza cognitiva, la disciplina per cui "spiegare un certo aspetto del comportamento di un sistema vivente significa [...] descrivere *il meccanismo mentale di elaborazione delle informazioni* che lo produce" [Datteri, 2017, p.8 enfasi originale]. Alla costituzione della Scienza Cognitiva, l'Intelligenza Artificiale contribuisce attraverso legami "forti" e "deboli" con altre aree del sapere, quali Psicologia, Filosofia, Neuroscienze, Antropologia e Linguistica [Gardner, 1985, p.51, trad. it.].

Due “assunti centrali” ne definiscono lo statuto metodologico e si riflettono direttamente con l'impianto dell'IA [Gardner, 1985, p.52].

Il primo è il *ruolo decisivo delle rappresentazioni*: la scienza cognitiva lavora "a un livello di analisi separato", quello della rappresentazione, situato "in una posizione intermedia tra input e output". L'attività cognitiva umana può essere descritta scientificamente "nei termini di simboli, di schemi, di immagini, di idee e di altre forme di rappresentazione mentale" [Gardner, 1985, pp.52-54 trad. it.].

Il secondo assunto è la *centralità del calcolatore*. Il computer, l'elaboratore artificiale di informazioni, ha avuto un ruolo chiave nella scienza cognitiva, poiché "assolve in primo luogo la funzione di 'prova di esistenza'": se una macchina può ragionare, avere obiettivi, correggere il proprio comportamento ed elaborare le informazioni, allora anche l'uomo può essere descritto negli stessi termini. Inoltre, il computer è un "prezioso strumento per l'analisi dei dati e la simulazione di processi cognitivi", al punto che l'intelligenza artificiale è considerata "la disciplina centrale della scienza cognitiva". Il lavoro con il computer è onnipresente e rappresenta, "malgrado le critiche", un "indicatore attendibile del legame di una disciplina con essa" [Gardner, 1985, trad.it, pp.54-56]²³⁷.

Nel capitolo precedente abbiamo analizzato come McCulloch e Pitts²³⁸ abbiano elaborato una “fisiologia dell'intelletto” [Arbib, 2003, p.11], approfondendo la scoperta di Shannon dell'equivalenza formale tra circuiti e i principi logici booleani. In *A Logical Calculus*, i due autori mostravano come i neuroni implementavano computazionalmente le operazioni della logica di prim'ordine, cioè implementavano, al pari di ogni sostrato capace di realizzarne l'organizzazione funzionale, “la psicologia del computabile” [Arbib, 2003, p.11]²³⁹ fornita da

²³⁷ Per la differenza dell'impiego del calcolatore tra l'IA (e le scienze cognitive classiche) e il connessionismo si veda Parisi, 1991, pp.9-10.

²³⁸ Non è certo se Turing fosse a conoscenza del lavoro di McCulloch e Pitts. Copeland suggerisce che probabilmente Wiener ne avesse parlato con Turing durante un incontro al NPL nel 1947 e che Turing, indirettamente, tramite *First Draft of a Report on the EDVAC* di von Neumann del 1945 [von Neumann, 1945], avesse sentito parlare di *A Logical Calculus* [Copeland, 2004cc, p.408], come dimostrano alcuni schemi logici della proposta per l'ACE. Tuttavia, è probabile che questo non abbia influenzato direttamente in alcun modo il suo lavoro. Anzi, dopo il loro primo, breve incontro nel 1949, Turing definì McCulloch “un ciarlatano” [Copeland, 2004cc, p.408].

²³⁹ “As one of the ingredients of Church's thesis, Turing offered a “psychology of the computable,” making plausible the claim that any effectively definable computation, that is, anything that a human could do in the way of symbolic manipulation by following a finite and completely explicit set of rules, could be carried out by such a machine equipped with a suitable program [...] “*A Logical Calculus of the Ideas Immanent in Nervous Activity*,” McCulloch and Pitts (1943) united the studies of neuro physiology and mathematical logic. Their formal model of the neuron as a threshold logic unit [...] built on the neuron doctrine of Ramon y Cajal and the excitatory and inhibitory synapses of Sherrington, using notation from the mathematical logic of Whitehead, Russell, and Carnap. McCulloch and Pitts provided the “physiology of the computable” by showing that the control box of any Turing machine, the

Turing. Inoltre, si è visto come la teoria e la manipolazione dell'informazione fosse centrale nel programma della cibernetica.

Turing compie un passo ulteriore, ovvero “aprire la strada a un ampliamento epocale della tipologia degli automi considerati agli esordi della cibernetica, portando in primo piano i calcolatori digitali *general purpose*” [Cordeschi, Tamburrini, 2015], cioè le macchine logiche digitali universali.

L'Intelligenza Artificiale simbolica, e con essa la scienza cognitiva classica che a quest'ultima si ispira, facendo leva sulla centralità del computer, si collocano – come affermano A. Newell e H. Simon in *Computer Science as Empirical Enquiry: Symbols and Search* [Newell, Simon, 1976] – in una prospettiva che assume come nucleo la manipolazione di strutture simboliche. Esse si inseriscono così nello stesso solco da cui ha preso avvio la nostra indagine: quello della formalizzazione totale della matematica proposta da Hilbert.

Le radici dell'ipotesi risalgono al programma di Frege e di Whitehead e Russell volto a formalizzare la logica: catturare le nozioni concettuali di base della matematica all'interno della logica e porre i concetti di dimostrazione e deduzione su basi solide. Questo sforzo culminò nella logica matematica - le nostre note logiche proposizionali, del primo ordine e di ordine superiore. Si sviluppò una visione caratteristica, spesso definita come il “gioco dei simboli”. La logica, e con essa tutta la matematica, era un gioco condotto con segni privi di significato secondo determinate regole puramente sintattiche. Ogni significato era stato eliminato. Ci si trovava di fronte a un sistema meccanico, sebbene permissivo (oggi diremmo non deterministico), all'interno del quale era possibile dimostrare varie proposizioni. I progressi furono dunque ottenuti allontanandosi da tutto ciò che sembrava pertinente al significato e ai simboli umani. Potremmo chiamare questa fase la fase della manipolazione formale dei simboli [Newell, Simon, 1976, p.88 trad. mia]²⁴⁰.

Inoltre la scarnificazione della semantica alla pura sintattica alla pura semantica, l'astrazione di ogni riferimento era il cuore della teoria dell'informazione proposta da Claude

essential formalization of symbolic computation, could be implemented by a network (with loops) of their formal neurons”.

²⁴⁰ “The roots of the hypothesis go back to the program of Frege and of Whitehead and Russell for formalizing logic: capturing the basic conceptual notions of mathematics in logic and putting the notions of proof and deduction on a secure footing. This effort culminated in mathematical logic-our familiar propositional, first-order, and higher-order logics. It developed a characteristic view, often referred to as the “symbol game”. Logic, and by incorporation all of mathematics, was a game played with meaningless tokens according to certain purely syntactic rules. All meaning had been purged. One had a mechanical, though permissive (we would now say nondeterministic), system about which various things could be proved. Thus progress was first made by walking away from all that seemed relevant to meaning and human symbols. We could call this the stage of formal symbol manipulation”.

Shannon. I calcolatori *general purpose* secondo questa teoria altro non sarebbero dei manipolatori di simboli secondo regole. Assume senso un assunto della GOFAI, quello definito da J. Haugeland “*il motto del formalista*”, e cioè: “Se tu ti occuperai della sintassi, la semantica si prenderà cura di sé stessa” [Haugeland, 1985, p.101 trad. it.].

È centrale in questa idea della pura manipolazione simbolica la stessa pretesa di totale trasparenza, di visibilità integrale, delle relazioni sintattiche secondo “regole” ben precise, che animava le intenzioni del programma di Hilbert.

Ma è sufficiente? “L’IA che coinvolge l’elaborazione simbolica semanticamente trasparente (*semantically transparent symbol processing*) può essere identificata con ciò che chiamo la visione dell’occhio della mente (*Mind’s Eye*)” [Clark, 1989, p.3 trad. mia]²⁴¹. Al pari delle macchine non organizzate di Turing che rappresentavano il tentativo del matematico inglese superare i limiti della Macchina Universale per sviluppare in esse un’intelligenza paragonabile a quella umana, “i “computer programmati non avevano nulla che ricordasse una vera intelligenza. Erano rigidi e fragili, capaci di eseguire solo alcuni compiti in modo interessante. Questo approccio [...] potrebbe aver sbagliato nel riporre troppa fiducia nella visione che la mente ha di sé stessa” [Clark, 1989, p.4 trad. mia]²⁴².

La stessa pretesa di trasparenza, o meglio, di autotrasparenza, del *Mind’s Eye*, come abbiamo visto viene meno in Turing, sia nelle macchine *oracolo* e sia a proposito dell’addestramento, il processo da egli individuato per giungere al pensiero intelligente, come nel brano sopra citato: “*Un aspetto importante delle macchine che apprendono è che spesso il loro istruttore saprà assai poco di che cosa succede esattamente là dentro*”.

Vi è uno scarto che attraversa, a più livelli, l’intero pensiero di Turing: da *On Computable Numbers*, dove la formalizzazione si ritorceva contro sé stessa nell’analisi dei problemi indecidibili, al contrasto tra *intuition* e *ingenuity* nella sua tesi di dottorato, sino alla distinzione tra “disciplina” e “iniziativa” nelle riflessioni sulle possibilità delle sue macchine di descrivere il funzionamento del cervello.

Ed è proprio l’“occhio del cervello” [Clark, 1989, p.4]²⁴³ quello che probabilmente sfugge allo sguardo autoriferito della mente su sé stessa. Quello, cioè, rappresentato dalle architetture

²⁴¹ “AI that involves conventional, semantically transparent symbol processing is to be identified with what I am calling the mind’s-eye view”.

²⁴² “The programmed computers lacked the smell of anything like real intelligence. They were rigid and brittle, capable of doing only a few tasks interestingly well. This approach (which was not universal) may have erred by placing too much faith in the mind’s own view of the mind”

²⁴³ ““brain’s-eye view.” The label refers to the brainlike structure of connectionist architectures. Such architectures are neurally inspired”.

connessioniste ispirato alle reti neurali. Ecco perché è stato importante mettere in luce gli aspetti in cui il pensiero di Turing, oltre ad ispirare il programma di ricerca dell'IA, ha anticipato il paradigma connessionista. Oltre alle caratteristiche, ispirate alla struttura neurofisiologica del cervello, di elasticità, plasticità, gestione dell'errore sopra notate, ve n'è un'altra particolarmente importante, la dimensione del parallelismo tra le varie reti neurali (artificiali o naturali).

Le reti neurali presenti in lumache, criceti, scimmie e umani sono infatti vasti network paralleli di processori relativamente lenti e semplici, ma fortemente interconnessi. La relativa lentezza dei singoli processori è compensata dal fatto che essi lavorano in modo cooperativo e parallelo sul compito da svolgere [...] Nell'informatica, questo tipo di cooperazione parallela si dimostra particolarmente utile quando un compito implica la soddisfazione simultanea di numerosi piccoli vincoli "morbidi". In questi casi-come la visione biologica e il controllo sensomotorio-un'architettura parallela cooperativa permette di affrontare compiti che, con processori così lenti, non sarebbero altrimenti risolvibili nei tempi disponibili. [...] Il segreto della vera intelligenza potrebbe essere rivelato proprio dalle lezioni che questi semplici organismi possono insegnarci. Potrebbe infatti darsi che la flessibilità e il senso comune che associamo alle conquiste cognitive umane derivino da una forma computazionale selezionata dall'evoluzione proprio per risolvere problemi di base legati alla percezione e al movimento [Clark, 1989, pp.4-5, trad.mia]²⁴⁴.

Così come dal programma logicista iniziato da Frege sono sorti dei "mostri" -i paradossi di Russell-e dai *Principia Mathematica* e dal Programma di Hilbert è nato l'Enunciato Gödeliano e l'*Entscheidungsproblem* ha generato una macchina, da questa macchina, diventata universale e *general purpose* e implementatasi nei calcolatori costruiti secondo l'"architettura di Von Neumann", emerge un modello di cervello.

Vedremo più avanti, nella comparazione tra "automi" biologici e artificiali - punto cruciale della metafora cibernetica - come il riconoscimento della natura probabilistica della

²⁴⁴ "The neural networks found in slugs, hamsters, monkeys, and humans are likewise vast parallel networks of richly interconnected, but relatively slow and simple, processors. The relative slowness of the individual processors is offset by having them work in a cooperative parallelism on the task at hand. [...] In computing, this kind of cooperative parallelism proves most useful when the task involves the simultaneous satisfaction of a large number of small, or "soft," constraints. In such cases (biological vision and sensorimotor control are prime examples) the use of a parallel cooperative architecture can make tractable tasks that (with such slow processors) we could not otherwise complete in the time available. [...] The secret of real intelligence may be revealed by the lessons such humble creatures are able to teach us. For it may be that the flexibility and common sense associated with human cognitive achievements is the result of an underlying computational form chosen by natural selection precisely for its ability to help solve such evolutionarily basic problems".

teoria dell'informazione di Shannon introduca, nel cuore stesso di questo design del calcolatore, assunto come modello della mente per l'IA e la scienza cognitiva classica, un oltrepassamento della logica che lo struttura e l'emergere di architetture computazionali più vicine alla conformazione del sistema nervoso.

Questo modello non è un'istanza irriducibile di natura mistica o metafisica, ma resta, come già osservato, pienamente computazionale. È in questa prospettiva che il pensiero di Turing diventa fondamentale: esso mostra come, nonostante le differenze di paradigma nell'IA e nelle scienze cognitive, e nonostante il ripresentarsi dello scarto già incontrato a livello delle facoltà, al livello di descrizione del cervello offerto dalle sue macchine non organizzate “presumibilmente, il comportamento intelligente consiste in un allontanamento dal comportamento in tutto e per tutto disciplinato in cui consiste un calcolo”, ma che tale allontanamento sia “piuttosto piccolo, che non dà origine a un comportamento casuale né a circolarità [loops] ripetitive del tutto prive di senso”.

A conferma di ciò vi è il suo tentativo di addestrare in questa direzione le macchine di tipo P, così come quello che abbiamo definito Principio di Turing: il pensiero va ascrivito a ogni descrizione computazionale del cervello, una descrizione che comprende elasticità, plasticità e architettura in parallelo.

Possiamo a questo punto applicare al nostro percorso e all'arco appena indicato, in modo simile a quanto fatto da Clark nei confronti dei sistemi semanticamente trasparenti (STS)²⁴⁵ [Clark, 1989, p.18]-ossia quei sistemi computazionali la cui struttura formale interna corrisponde chiaramente al livello concettuale, rendendo così le loro operazioni interpretabili, e a sua volta ispirati dal programma logicista e da quello di Hilbert-il metodo di studio dei sistemi complessi di elaborazione dell'informazione del neuroscienziato e fisiologo britannico David Marr (1945-1980) nel suo studio *Vision. A Computational Investigation into the Human Representation and Processing of Visual Information*, uscito postumo nel 1982 [Marr, 2010].

Marr pone tre livelli di analisi. Il primo livello (1) è quello della teoria computazionale (*Computational theory*), che si chiede: “Qual è l'obiettivo della computazione, perché è appropriato e qual è la logica della strategia con cui può essere eseguito?” [Marr, 2010, p.25 figure 1-4 trad. mia]²⁴⁶. Questo è il livello più astratto. Concerne la specificazione del compito

²⁴⁵ “A system will be said to be semantically transparent just in case it is possible to describe a neat mapping between a symbolic (conceptual level) semantic description of the system's behavior and some projectible semantic interpretation of the internally represented objects of its formal computational activity”.

²⁴⁶ “What is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out?”.

associato rispetto a un compito cognitivo [Frixione, Palladino, 2011, p.137] ad un livello di corrispondenza puramente funzionale tra input e output [Frixione, Palladino, 2011, p.137]. Il livello computazionale descrive l'obiettivo della computazione, le strategie generali per raggiungerlo e i vincoli su tali strategie [Clark, 1989, p.18] e cioè cosa fa il sistema [Datteri, 2017, p.89].

Il secondo livello (2), quello centrale, è quello della rappresentazione e dell'algoritmo (*Representation and algorithm*), che risponde alle domande "Come può essere implementata questa teoria computazionale? In particolare, qual è la rappresentazione per l'input e l'output e qual è l'algoritmo per la trasformazione?" [Marr, 2010, p.25 figure 1-4 trad. mia]²⁴⁷. Questo livello riguarda la rappresentazione scelta per indicare l'input e l'output, insieme all'algoritmo che trasforma il primo nel secondo. Esso corrisponde al modo in cui un determinato compito viene eseguito [Frixione, Palladino, 2011, p. 138] e consente di individuare un meccanismo di elaborazione delle informazioni capace di spiegare il comportamento in esame [Datteri, 2017, p. 89].

Il terzo livello è quello dell'implementazione hardware (*Hardware implementation*), che corrisponde alla domanda "Come possono la rappresentazione e l'algoritmo essere realizzati fisicamente?" [Marr, 2010, p.25 figure 1-4 trad. mia]²⁴⁸. Questo livello riguarda la struttura del calcolatore, la realizzazione (sia essa biologica oppure elettronica, se si tratta di una simulazione al calcolatore) di un meccanismo [Datteri, 2017, p.89-90].

Questi tre livelli sono "interconnessi, ma solo debolmente. Ad esempio, la scelta di un algoritmo è influenzata da ciò che deve fare e dall'*hardware* su cui deve essere eseguito. Tuttavia, a ogni livello è possibile scegliere tra un'ampia varietà [Marr, 2010, p.25].

Inoltre, la relazione che sussiste tra (1) e (2) è diversa da quella che si instaura tra (2) e (3). La prima relazione (1)-(2) è assimilabile all'analisi meccanicistica: si identifica il meccanismo responsabile della produzione di una certa relazione ingresso-uscita [Datteri, 2017, p.90]. La seconda relazione (2)-(3) è quella che si propone di identificare la realizzazione di un certo meccanismo di elaborazione delle informazioni, significa formulare una buona spiegazione neuroscientifica (o elettronica) di un comportamento che era già stato soggetto a un'analisi meccanicistica in termini psicologici [Datteri, 2017, p.90].

²⁴⁷ "How can the representational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation?".

²⁴⁸ "How can the computation and algorithm be realized physically?".

Nella teoria di Marr, i livelli di analisi e le loro relazioni possono consentire, a mio avviso, un ampliamento concettuale che integri la descrizione neurofisiologica all'interno del computazionalismo, in continuità con l'analisi di Turing qui sviluppata, secondo una duplice prospettiva bottom-up e top-down. La prima, orientata verso l'alto, individua un livello di spiegazione più astratto rispetto alla formulazione di algoritmi o alla scrittura di programmi per calcolatore [Frixione, Palladino, 2011, p. 139]; la seconda, orientata verso il basso, riconosce un ruolo ai dati delle neuroscienze, tradizionalmente emarginati dalla metodologia funzionalista come non pertinenti [Frixione, Palladino, 2011, p. 139]²⁴⁹.

Si introduce così una distinzione tra la computazione e l'algoritmo corrispondente, che al tempo stesso include la dimensione neuronale e cerebrale concreta. A mio avviso, la tripartizione dei livelli di analisi sopra richiamata può essere applicata anche all'itinerario del pensiero di Turing qui preso in esame. Si osserva infatti come, a partire da un sistema computazionale astratto definito da vincoli rigorosi - le regole metamatematiche del formalismo hilbertiano - si giunga a un livello di specificazione algoritmica, rappresentato dalla Macchina di Turing, che a sua volta può essere implementata, in maniera non rigida, in un hardware: sia quello di un calcolatore, sia quello, dalla consistenza di *pudding*, del cervello umano.

Da un lato, ciò consente di adottare nella prospettiva *top-down* un'impostazione computazionale in scienza cognitiva che non risulta incompatibile con l'attenzione ai dati del sistema nervoso e ai risultati delle neuroscienze [Frixione, Palladino, 2011, p. 139]; in altri termini, riflette la tensione costante in Turing verso la descrizione del cervello e delle sue facoltà. Dall'altro lato, nella prospettiva *bottom-up*, il computazionalismo non resta vincolato a un'architettura di calcolo specifica, come quella sequenziale ispirata al modello di von Neumann, ma può essere - con la necessaria prudenza metodologica - interpretato come compatibile anche con le architetture parallele tipiche del connessionismo, di cui Turing può essere considerato, come si è visto, un pioniere.

È possibile prendere in esame un esempio in cui il lavoro di Turing riflette la tensione che emerge nelle analisi precedenti tra apprendimento delle macchine, la dimensione di formalizzazione computazionale di funzioni ascrivibili al pensiero e dimensione implementativa. Abbiamo visto come nella versione embrionale dell'*Imitation Game* porposta

²⁴⁹ Sul rapporto e le differenze tra i livelli di analisi Marr e il connessionismo si veda Churchland, Sejnowski, 1990.

in *Intelligent Machinery*, in cui immaginava una *paper machine* che giocasse “non troppo male” impegnata in partite con altri giocatori mediocri di scacchi.

Turing realizzò effettivamente, nell'estate del 1948, la macchina di carta in questione in collaborazione con il suo amico, il matematico statistico David Champernowne (1912 – 2000). Questa era “una sorta di programma in pseudo-codice - un insieme di regole di macchina per giocare a scacchi”, chiamata *Turochamp*. Questa *paper machine* – rimasta su carta, e simulata con “carta e matita” perché nel 1948 non c'erano ancora calcolatori su cui implementarla – “fu una delle prime manifestazioni di quello che oggi chiameremmo un programma di intelligenza artificiale” [Copeland, Prinz, 2017, p. 331 trad. mia]²⁵⁰.

Turing provò a implementare *Turochamp* nel calcolatore di Manchester (che analizzeremo in seguito) quando assunse il suo ruolo ma non riuscì a completare il compito, al contrario del gioco della dama che fu effettivamente realizzato nella macchina del Manchester Manchester Computing Machine Laboratory, diretto da Max Newman, dal suo amico e collaboratore C. S. Strachey.

Al contrario di un programma a “forza bruta” [*brute force*], che “seleziona le sue mosse esplorando le conseguenze di ciascuna mossa a sua disposizione” [Copeland, Prinz, 2017, p.332 trad. mia]²⁵¹, “*Turochamp* poteva, in linea di principio, giocare una partita completa e non operava tramite una ricerca esaustiva, ma sotto la guida di euristiche” [Copeland, 2004d, p.564]. Questo concetto è particolarmente importante nello sviluppo dell'IA classica e che si attribuisce solitamente al *Logic Theorist* sviluppato in occasione della conferenza di Dartmouth da Newell e Simon nel 1956 [McCorduck, 2004, xxvi]²⁵².

²⁵⁰ “The ‘paper machine’ that Turing mentioned was created that same summer in collaboration with his friend the statistician David Champernowne. Turing crystallized his ideas about computational chess into a kind of program in pseudo-code-a set of machine-rules for playing chess. Named ‘Turochamp’, and an outgrowth of his wartime deliberations with Michie and Good, this was an early manifestation of what would now be called an AI program. Turochamp was a ‘paper machine’ in the sense that since no electronic computer was available, Turing and Champernowne simulated the program’s play by hand, using paper and pencil”. La descrizione delle regole di funzionamento di *Turochamp* è data dallo stesso Champernowne in Copeland, Prinz, 2017, p.331.

²⁵¹ “A brute-force chess program selects its moves by exploring the consequences of each move available to it”.

²⁵² Sul *Logic Theorist*, che fu capace di risolvere 38 dei 52 problemi del secondo capitolo dei *Principia Mathematica* trovando prove più semplici per alcuni di questi [Newell, Simon, 1956, p.79], si veda McCorduck, 2004, pp.161-170; Boden, 2006, 323-338. Per il funzionamento del programma di veda Gugerty, 2006 e il report elaborato dagli stessi Newell e Simon per la RAND Corporation, Newell, Simon, 1956b. Newell e Simon presentarono i risultati della loro ricerca al Symposium sull'Information Theory del MIT del 10-12 settembre – l'atto di nascita della scienza cognitiva- mostrando come la ricerca euristica, il cuore del *Logic Theorist*, fosse un sistema di “problem solving”, ispirato alla capacità di risoluzione dei problemi complesse dell'essere umano - in opposizione a quelli, algoritmici, che richiedono solo una gran quantità di elaborazione - visto anch'esso come sistemi di information processing [Newell, Simon, 1965a, pp.61-79]. Sul concetto di euristica per i due pionieri dell'IA si veda Newell, Simon, 1958.

Turing condensò, con l'esperienza maturata nei tentativi di implementazione in hardware fatti, le sue idee sugli scacchi nel saggio intitolato *Chess* [Turing 1953, in Copeland, 2004d, pp. 569-575]²⁵³, pubblicato inizialmente come sezione del capitolo "*Digital Computers Applied To Games*", con co-autori Audrey Bates, Bowden e lo stesso Strachey, nel volume *Faster Than Thought*, che offriva una sorta di rassegna dello stato dell'arte degli sviluppi della computazione digitale ad inizio anni '50 [Bowden, 1953].

In tale lavoro, Turing prende le mosse da un assunto che intende confutare, ossia l'idea secondo cui "non si può programmare una macchina a giocare una partita migliore di quanto non si giochi personalmente", concetto che egli paragona a quello per cui "nessun animale può inghiottire un animale più pesante di sé stesso". Il suo giudizio su entrambe le affermazioni è netto: "Entrambe le affermazioni sono, per quanto ne so, false". Come egli osserva, infatti, per smentire questa credenza "è sufficiente molto semplicemente [considerare] la velocità della macchina, che potrebbe rendere possibile spingere l'analisi una mossa più avanti di quanto un uomo riuscirebbe a fare nello stesso tempo" [Turing, 1953, p. 574-575 trad. mia]²⁵⁴.

Il programma scacchistico di Turing anticipava molte delle tecniche che sarebbero diventate standard nella programmazione: l'impiego di euristiche per orientare la ricerca all'interno dell'albero delle mosse e contromosse; l'applicazione di regole di valutazione capaci di attribuire valori numerici alle configurazioni della scacchiera in base alla loro forza o debolezza; l'adozione della strategia del minimax; e l'uso di una previsione variabile, che permetteva di approfondire maggiormente l'analisi delle mosse più promettenti rispetto a quelle meno vantaggiose [Copeland, 2004d, p. 565]. Sul piano dell'apprendimento, Turing suggeriva che la macchina sperimentasse variazioni nel proprio metodo di gioco – ad esempio modificando i valori numerici attribuiti ai pezzi – e mantenesse quelle che conducevano a risultati più soddisfacenti. Egli riconosceva inoltre la necessità di adottare un sistema del tutto diverso per affrontare la fase finale della partita [Turing, 1953, pp. 571, 574]²⁵⁵.

²⁵³ Qui si farà riferimento al contributo di Turing riportato in Copeland, 2004

²⁵⁴ "This fact might be regarded as supporting the rather glib view which is often expressed, to the effect that 'one cannot programme a machine to play a better game than one plays oneself'. This statement should I think be compared with another of rather similar form. 'No animal can swallow an animal heavier than himself.' Both statements are, so far as I know, untrue [...] The statement about chess programming may be falsified quite simply by the speed of the machine, which might make it feasible to carry the analysis a move farther than a man could do in the same time."

²⁵⁵ Turochamp fu il primo programma scacchistico a battere un giocatore umano, la moglie di Champernowne, Mieke.[Copeland, Prinz, 2017, p.332]. Il leggendario campione di scacchi russo Garry Kasparov ha battuto Turochamp nel giugno del 2012 a Manchester [Copeland, Prinz, 2017, pp.327-329].

L'approccio di Turing al problema è in continuità con le riflessioni elaborate durante il periodo di Bletchley Park sul tema degli scacchi, un tema che lo aveva appassionato sin dall'infanzia [Hodges, 1985, pp. 278-279, trad. it.]. Già nel 1937, a Princeton, egli aveva assistito alle conferenze di von Neumann, dedicate alla teoria dei giochi e alla dimostrazione, risalente al 1928, dell'esistenza di strategie ottimali per ogni gioco a due persone con regole fisse [von Neumann, 1928; Hodges, 1985, p. 279, trad. it.]. La strategia in questione è quella del "minimax", secondo la quale, nei giochi a somma zero, ciascun giocatore cerca di massimizzare il proprio guadagno riducendo al minimo quello dell'avversario [von Neumann, 1928; Israel, Gasca, 2008 pp.66-80; Leonard, 2010, pp.56-76].

Gli scacchi costituiscono un caso esemplare di gioco a somma zero, "in cui la vincita di un giocatore è uguale e di segno opposto a quella del secondo" [Israel, Gasca, 2008, p. 70]. Essi rappresentano inoltre, come la dama, "un sistema completo di regole che dettano il da farsi in ogni contingenza" [Hodges, 1985, p. 280, trad. it.]. In questo *micromondo* rigorosamente normato, i giocatori operano in condizioni di "informazione perfetta" [Morgenstern, von Neumann, 1944, p. 124], poiché tutte le mosse possibili e le conseguenze delle azioni sono esplicitate e rese visibili a priori dal sistema di regole

Molto simili alle idee di Turing espresse in *Chess* furono quelle di Samuel sulla dama [Samuel, 1959], nel cui articolo compare per la prima volta l'espressione *machine learning*. Il suo programma, sviluppato per l'IBM 704, si basava sulla strategia del minimax e valutava le posizioni mediante un polinomio che combinava criteri come vantaggio di pezzi, mobilità e controllo del centro. Samuel sperimentava inoltre due forme di apprendimento: il "*rote learning*", che memorizzava le posizioni già incontrate, e il "*generalization learning*", che modificava progressivamente i coefficienti di valutazione in base all'esperienza di gioco [Samuel, 1959]²⁵⁶. In questo modo la macchina mostrava la possibilità di migliorare la propria

²⁵⁶ Per un'analisi del lavoro di Samuel sul suo programma per il gioco della dama si rimanda a Boden, 2006, pp. 713-715. "I programmi di Samuel impararono a giocare a un livello da forte dilettante. In tal modo egli confutò l'idea secondo cui i computer possono fare solo ciò che viene loro detto: il suo programma imparò rapidamente a giocare meglio del proprio creatore. Il programma fu dimostrato in televisione nel 1956, suscitando una forte impressione. Come Turing, anche Samuel ebbe difficoltà a trovare tempo di calcolo" ["Samuel's programs learned to play at a strong amateur level. He thereby disproved the idea that computers can do only what they are told to: his program quickly learned to play a better game than its creator. The program was demonstrated on television in 1956, creating a strong impression. Like Turing, Samuel had trouble finding computer", Russell, Norvig, 2021, 1.3.2]. Roberto Cordeschi fa notare come Samuel avesse già compreso che i giochi costituivano soltanto una prima tappa verso la comprensione dei metodi di simulazione del comportamento intelligente, prevedendo che in futuro tali metodi sarebbero stati applicati "sempre più spesso alle situazioni della vita reale", mentre l'attenzione ai giochi e ad altri *toy-problems* sarebbe progressivamente diminuita. A più di trent'anni di distanza, la sua previsione si è dimostrata sostanzialmente corretta. È vero che i *toy-problems* e la ricerca sui micromondi) hanno continuato a svolgere un ruolo rilevante nella ricerca in IA, analogamente a quanto la *Drosophila* ha rappresentato per la

prestazione attraverso il confronto con l'ambiente, un'idea che Wiener salutò come "fondamentale" nella prefazione alla seconda edizione di *Cibernetica*, a proposito dello sviluppo di macchine capaci di apprendere dai feedback ricevuti [Wiener, 1961, p.17 trad. it.]²⁵⁷. La portata di questo approccio si vide nel 1962, quando il programma di Samuel riuscì a sconfiggere il maestro di dama Robert Nealey, in quello che fu ricordato come uno dei primi successi di un computer contro un giocatore esperto.

Su questa stessa linea si collocano anche i tentativi di Shannon di costruire macchine capaci di giocare a scacchi [Shannon, 1950]. "Shannon pensava fin dal 1945 di meccanizzare il gioco degli scacchi mediante una strategia 'minimax', cioè in sostanza con una ricerca lungo l'albero delle mosse: la stessa idea di fondo che Alan e Jack Good avevano formalizzato nel 1941" [Hodges, 1985, pp.467-468 trad. it.]. L'ingegnere dei Bell Labs, discutendo degli automi universali di Turing programmabili per imparare a giocare, scriveva – a proposito del programma di dama implementato sulla macchina di Manchester, che incontreremo più avanti:

Il gioco della dama è stato recentemente programmato su un calcolatore di impiego generale, adoperando un metodo del tipo 'principio generale'. Strachey adoperò un metodo simile a quello proposto da chi scrive per programmare il gioco degli scacchi, svolgendo un'indagine delle possibili varianti di alcune mosse e una valutazione "minimax" (cioè basata sulla minimizzazione del massimo rischio di perdita) delle posizioni risultanti [Shannon, 1953, p.101 trad. it.].

Il riferimento di Shannon agli scacchi non è casuale. Egli infatti osserva che, considerando un computer elettronico (una macchina di Turing realizzata), "a patto di sostituire 'dentro la macchina' a 'mentalmente' [si tengano presenti gli 'stati mentali' di *On Computable Numbers*]" si ottiene esattamente una definizione di pensiero. Una definizione che Shannon attribuisce a "certi psicologi" (il nascente paradigma cognitivista) i quali lo descrivono come "essenzialmente caratterizzato dai seguenti passaggi: varie soluzioni possibili a un problema vengono provate mentalmente o simbolicamente, senza essere effettivamente

genetica, fornendo modelli sperimentali semplici e controllabili; ma il nucleo metodologico rimaneva legato alla ricerca euristica, cioè all'individuazione di strategie capaci di restringere lo spazio delle alternative esplorabili entro limiti di tempo e risorse computazionali accettabili. Tuttavia, l'interesse dei ricercatori si è progressivamente spostato verso i problemi *real-life*, dove la difficoltà non risiede soltanto nell'efficienza della ricerca, ma soprattutto nella rappresentazione della conoscenza necessaria per affrontare spazi complessi. [Cordeschi, 1994, pp.38-39]

²⁵⁷ "La nozione di macchina ad apprendimento può essere estesa ben oltre il suo impiego per previsori, filtri, e altri analoghi apparati. Essa è particolarmente importante per lo studio e la costruzione di macchine che svolgano giochi competitivi come la dama. In questo campo, il lavoro fondamentale è stato compiuto da Samuel e Watanabe nei laboratori della IBM".

realizzate fisicamente; la soluzione migliore viene selezionata tramite una valutazione mentale dei risultati di queste prove; e la soluzione così trovata viene poi messa in atto” [Shannon, 1950, trad. mia]²⁵⁸.

Le idee di Turing, espresse in *Chess*, sembrano essere in linea con questa prospettiva, e con il procedimento di modellizzazione computazionale degli stati mentali di *On Computable Numbers*. Scrive Turing, a proposito del sistema, “Lo descriverei come una caricatura del mio stesso gioco. Era infatti basato su un’analisi introspettiva dei miei processi di pensiero durante la partita, con considerevoli semplificazioni” [Turing, 1953, p.574 trad. mia]²⁵⁹. Secondo Turing, le regole delineate in *Chess* derivano da un’analisi introspettiva dei processi mentali che egli stesso metteva in atto nel corso del gioco [Copeland, 2004d, p. 565].

Stando a quanto emerge sinora dal suo lavoro sugli scacchi, il pensiero di Turing sembra giustificare l’assunzione, da parte del paradigma simbolico dell’IA e delle scienze cognitive, della sua macchina come modello della mente, come suggeriva del resto anche la riflessione di Shannon appena considerata. In una definizione “classica” di questo approccio, i filosofi e neuroscienziati Jerry Fodor (1935–2017) e Zenon Pylyshyn (1937–2022) scrivono:

(1) «Le teorie classiche - ma non quelle connessioniste - postulano un “linguaggio del pensiero”.» Ciò significa che esse postulano rappresentazioni mentali (strutture dati) con una certa forma. Tali rappresentazioni hanno una struttura sintattica, vale a dire sono costruite sistematicamente combinando costituenti atomici in aggregati molecolari, i quali (nei casi complessi) a loro volta formano intere strutture dati. In breve, esse postulano sistemi simbolici con una sintassi e una semantica combinatoria.

(2) «Nei modelli classici, i principi attraverso i quali gli stati mentali vengono trasformati, o mediante i quali un input seleziona il corrispondente output, sono definiti sulle proprietà strutturali delle rappresentazioni mentali. Poiché le rappresentazioni mentali classiche hanno una struttura combinatoria, è possibile che le operazioni mentali classiche si applichino ad esse facendo riferimento alla loro forma.» Questo significa che, se si dispone di un certo tipo di rappresentazioni strutturate (come richiesto al punto 1), è possibile definire operazioni computazionali su tali rappresentazioni in modo che esse siano sensibili a quella struttura. Se tale struttura non esiste (cioè, se non c’è una

²⁵⁸ “The thinking process is considered by some psychologists to be essentially characterized by the following steps: various possible solutions of a problem are tried out mentally or symbolically without actually being carried out physically; the best solution is selected by a mental evaluation of the results of these trials; and the solution found in this way is then acted upon. It will be seen that this is almost an exact description of how a chess-playing computer operates, provided we substitute “within the machine” for “mentally.”

²⁵⁹ “It was in fact based on an introspective analysis of my thought processes when playing, with considerable simplifications”.

rappresentazione simbolica), non si potrebbe farlo, anche se si potrebbe dare l'impressione di riuscirci fissando opportunamente una funzione in estensione [Fodor e Pylyshyn 1988, pp. 12–13 trad. mia]²⁶⁰.

Ma ci si può chiedere: è sufficiente questo procedimento di implementazione di strategie computabili, “scarnificate” e matematicamente modellizzabili, per attribuire alla macchina forme di apprendimento, intelligenza o pensiero?

Per rispondere a questa domanda, possiamo rifarci a quanto lo stesso Turing affermò, citando esplicitamente il lavoro di Shannon sugli scacchi, nella conferenza alla *London Mathematical Society* del settembre 1947, in cui presentava gli sviluppi dell'ACE (il calcolatore che aveva progettato al *National Physical Laboratory*, e che analizzeremo in seguito):

Sarebbe probabilmente facile escogitare tavole di istruzioni che permettano all'ACE di vincere contro un giocatore di medio livello. In effetti Shannon, dei laboratori della Bell Telephone, mi informa che ha vinto alcune partite giocando con regole empiriche. Non ho informazioni sull'abilità dell'avversario, ma non considererei tali vittorie molto significative. Ciò che vogliamo è una macchina che possa imparare dall'esperienza. Il permettere alla macchina di alterare le sue proprie istruzioni è un passo in questa direzione, ma naturalmente non ci fa andare molto avanti [Turing, 1947, p. 86, trad. it.].

Le idee di Turing sulla possibilità di attribuire il pensiero a macchine di questo tipo sembrano dunque diverse: egli voleva che le macchine imparassero dall'esperienza secondo regole non riducibili a semplici procedure empiriche formalizzate. Ritroviamo qui la stessa tensione verso il superamento del modello della Macchina Universale. Come far sì che questo avvenga?

Una risposta venne da Christopher Strachey, che – come vedremo – implementò un gioco della dama sulla macchina di Manchester, ben prima di Samuel, impiegando le regole

²⁶⁰ “(1) “Classical theories-but not connectionist theories-posit a ‘language of thought’.” This means that they posit mental representations (data structures) with a certain form. Such representations are syntactically structured, i.e., they are systematically built by combining atomic constituents into molecular assemblies, which (in complex cases) make up whole data structures in turn. In short, they posit symbol systems with a combinatorial syntax and semantics. (2) “In classical models, the principles by which mental states are transformed, or by which an input selects the corresponding output, are defined over structural properties of mental representations. Because classical mental representations have combinatorial structure, it is possible for classical mental operations to apply to them by reference to their form.” This means that if you have a certain kind of structured representations available (as demanded by point 1), it is possible to define computational operations on those representations so that the operations are sensitive to that structure. If the structure isn't there (i.e., if there is no symbolic representation), you couldn't do it, though you might make it look as if you had by fixing on a suitable function in extension”

esemplificate nel *Turochamp* ed espresse in *Chess*. Secondo Strachey, il primo compito, nel conseguimento dell'intelligenza – o delle facoltà del cervello umano – consiste nel “far sì che la macchina impari nel modo in cui impara un bambino, con l'aiuto di un insegnante”. Come Turing, egli sosteneva che una delle “caratteristiche più importanti del pensiero” è “l'imparare da soli attraverso l'esperienza, senza l'aiuto di un insegnante” [Proudfoot, 2017, p.317].

La chiave, dunque, era nelle *macchine non organizzate di tipo-P*: le *macchine-bambino* su cui Turing stava sperimentando, e nelle quali aveva immaginato un embrione dell'Imitation Game fondato proprio sugli scacchi. Esse rappresentavano un tentativo di superare i limiti della sola disciplina computazionale delle macchine universali e un cambiamento paradigmatico verso strutture più simili al cervello e alla sua struttura biologica neuronale, realizzabili in un calcolatore.

È necessario ora prendere in esame la storia dei tentativi di realizzazione in hardware delle idee di Turing: sia quelli che lo hanno visto direttamente protagonista, sia quelli che, sull'altra sponda dell'Atlantico, hanno portato alla costruzione delle macchine che avrebbero incarnato l'architettura di von Neumann, concepita come modello della mente umana e del rapporto con l'informazione. Prima di passare a questa analisi, occorre però un'ultima riflessione.

Scriva Turing in *Computing machinery and intelligence*:

I calcolatori digitali [...] possono essere classificati tra le «macchine a stati discreti», cioè tra quelle che si muovono a salti o scatti improvvisi da uno stato ben definito a un altro. Questi stati sono abbastanza differenti tra loro perché si possa ignorare la possibilità di confusione tra di essi. Strettamente parlando non esistono macchine di questo genere: in realtà ogni cosa si muove con continuità. Ma ci sono molti tipi di macchine che possono vantaggiosamente essere viste come macchine a stati discreti [Turing, 1950, p. 129 trad. it. 1994].

Sebbene Turing, nel rispondere alle obiezioni al gioco dell'imitazione, riconoscesse la continuità del sistema nervoso [cfr. Turing, 1950, p. 145 trad. it. 1994], dal punto di vista della sua analisi delle macchine “il cervello, come il computer, era una macchina a stati discreti, ovvero una diversa realizzazione fisica di una macchina di Turing universale. In altre parole, il computer era un cervello a un livello di astrazione più alto di quello identificato da Wiener” [Marconi, 2025].

La “proprietà peculiare dei calcolatori digitali, per cui sono in grado di imitare qualsiasi macchina a stati discreti, viene descritta dicendo che sono macchine universali” [Turing, 1950, p. 132 trad. it.]. È questa caratteristica che deve ora essere messa alla prova della realizzazione concreta, nel passaggio dalle formulazioni teoriche all’implementazione effettiva. Per comprenderla appieno, e prima di analizzare come essa sia stata incarnata nelle macchine reali – nell’*architettura di von Neumann* e nello *stored-program principle* – occorre chiarire il significato del concetto di “discreto” in opposizione a quello di “continuo”, alla luce delle considerazioni sulla cibernetica e sulla teoria dell’informazione, e il passaggio dal paradigma analogico a quello digitale, in rapporto sia alle condizioni storiche – le esigenze belliche - di sviluppo sia alle capacità computazionali dell’essere umano che queste macchine mirano a riprodurre o ad ampliare.

Cap. 3: Il visibile e l'invisibile

1. Istruzioni storiche per un cervello digitale

Le idee di Wiener, Shannon e degli altri pionieri della rivoluzione digitale, analizzate in precedenza, non nacquero come semplici speculazioni accademiche. La loro genesi, il loro sviluppo, e persino la forma concettuale che assunsero, furono profondamente intrecciati con l'implementazione concreta di macchine e strumenti, come nel caso dell'*Antiaircraft Predictor* o di SIGSALY, progettati per rispondere a esigenze specifiche del contesto bellico.

La Seconda Guerra Mondiale costituì un fattore decisivo per lo sviluppo della dimensione hardware, che dette concreta attuazione agli studi sulla cibernetica, sulla teoria dell'informazione e sulla computabilità, influenzandone e orientandone le successive evoluzioni. Il conflitto contro il nazifascismo fornì il contesto in cui tali concezioni poterono radicarsi, svilupparsi e riprodurre le logiche originarie, fino a trasformare la realtà socio-culturale e socio-tecnica che ancora oggi abitiamo.

Fondamentale fu, a questo proposito, il contributo di uno degli "scienziati di guerra" *par excellence*, John von Neumann²⁶¹, affinché le idee finanziate dai governi impegnati nella Seconda

²⁶¹ Sebbene il suo contributo alla teorizzazione e alla realizzazione dei calcolatori - progenitori di quelli odierni - sia stato assolutamente decisivo, la figura di John von Neumann è irriducibile a questo solo ambito. Nato Neumann János Lajos a Budapest nel 1903, allora parte dell'Impero austro-ungarico, in una famiglia alto-borghese di origine ebraica (il padre, Max [Maksi], fu insignito del titolo nobiliare von per meriti nel 1913 dall'Imperatore Francesco Giuseppe, da qui il prefisso "von"), fu sin da bambino un genio precocissimo. Scrisse il suo primo articolo scientifico all'età di 17 anni, insieme a Michael Fekete: *Über die Lage der Nullstellen gewisser Minimumpolynome* [Fekete, von Neumann, 1922], uno studio sugli zeri di particolari polinomi minimali, generalizzazione del teorema di Fejér sulla posizione delle radici dei polinomi di Tchebyscheff [Ulam, 1958]. A diciannove anni scrisse *Die Axiomatisierung der Mengenlehre*, poi pubblicata nel 1925 in forma più accessibile sostituendo il solenne "Die" con un più modesto "Eine". L'Interesse di von Neumann andava spostandosi attorno ai fondamenti della logica matematica e dei numeri naturali e per l'assiomatizzazione - volta ad evitare i paradossi di Russell - della teoria degli insiemi: una tensione alla formalizzazione che lo accompagnerà anche negli studi futuri sui calcolatori. Laureatosi in chimica a Zurigo e addottoratosi in matematica a Budapest nel 1926 nell'autunno dello stesso anno si trasferì a Göttingen - allora capitale mondiale della matematica - per studiare sotto la guida di David Hilbert. Intanto, diventava *Privatdozent* a Berlino - il più giovane nella storia dell'università della capitale tedesca - e ad Amburgo. I suoi interessi si spostarono sui fondamenti della fisica quantistica, culminando nella pubblicazione di *Mathematische Grundlagen der Quantenmechanik* [von Neumann, 1932]. Quest'opera divenne la teorizzazione "canonica" della matematica dell'interpretazione di Copenaghen. Qui von Neumann elaborò la celebre "catena di von Neumann", inserendo formalmente l'osservatore nel processo di misurazione e riconoscendo il ruolo cruciale della coscienza nel collasso della funzione d'onda. Nel 1929 si recò per la prima volta a Princeton come *visiting lecturer*, ritornando l'anno successivo come *visiting professor*. Nel 1931 ottenne una cattedra permanente e, nel 1933, fu chiamato come membro permanente all'*Institute for Advanced Study*, il più giovane tra i membri della facoltà. Con l'ascesa al potere del nazismo in Germania, von Neumann recise ogni legame con l'accademia tedesca. Nell'odio per il nazismo - che si sommava a quello verso il comunismo, maturato durante l'infanzia a Budapest, all'epoca della traumatica esperienza della Repubblica dei Consigli di Béla Kun nel 1919 - si può riconoscere uno dei motori principali del suo impegno come collaboratore del governo e dell'esercito degli Stati Uniti. Nel 1937, appena ottenuta la

Guerra Mondiale – che presero corpo nei calcolatori e, al tempo stesso, furono ispirate dalle nuove possibilità che queste macchine rendevano immaginabili – divenissero *visibili*: riconosciute, accessibili al pubblico e impiegabili nella ricerca scientifica e nell'uso commerciale²⁶².

cittadinanza statunitense, tentò di entrare nell'esercito come ufficiale: nonostante avesse ottenuto il massimo dei voti all'esame scritto, fu respinto per superamento del limite di età. Grazie all'interessamento di Oswald Veblen - uno dei "grandi vecchi" dell'Institute for Advanced Study, già leader, tra le altre cose, degli studiosi di balistica e idrodinamica negli USA e consulente di punta del governo americano nella Prima guerra mondiale - von Neumann fu nominato consulente scientifico per il Dipartimento della Difesa. Partecipò dapprima al comitato di consulenza del Laboratorio di Balistica dell'Esercito, poi al comitato sulla mobilitazione scientifica per la guerra, e infine al comitato scientifico della National Defense Research Committee. I suoi studi sulle onde d'urto - fondamentali per la teoria dei fluidi compressibili e per l'idrodinamica delle esplosioni - ebbero conseguenze dirette nello sviluppo di armamenti, in particolare nella massimizzazione degli effetti esplosivi. Fu questo interesse che portò von Neumann a essere coinvolto nel Progetto Manhattan, dove si occupò dell'ottimizzazione delle *lenti esplosive*, dispositivi fondamentali per ottenere l'implosione necessaria al corretto innesco delle bombe a fissione. La crescente esigenza di capacità computazionale nello studio di questi problemi lo condusse verso l'ENIAC, utilizzato per il calcolo delle traiettorie e dei parametri relativi alla bomba all'idrogeno in costruzione a Los Alamos. Da qui nacque l'idea di un suo superamento logico e architettureale, l'EDVAC, di cui von Neumann redasse il celebre *First Draft of a Report on the EDVAC* (1945), un testo fondativo dello *stored program principle*. Le riflessioni sui calcolatori lo spinsero anche ad approfondire le analogie e le differenze tra sistemi artificiali e biologici, conducendolo allo studio del sistema nervoso umano, studi culminati nel suo *Theory of Self-Reproducing Automata* che è considerato lo studio che ha aperto la via - oltre a gettare le basi concettuali per l'IA basata sulle reti neurali - a quel campo oggi noto come *Artificial Life*. Nell'ottobre del 1954 fu nominato dal presidente Eisenhower membro della *United States Atomic Energy Commission*. Nello stesso periodo fu invitato a tenere le prestigiose *Silliman Memorial Lectures* all'Università di Yale sul tema dei calcolatori e del cervello. Tuttavia, durante la preparazione delle lezioni, gli fu diagnosticato un cancro incurabile alle ossa. Gli appunti di quelle conferenze furono poi pubblicati postumi nel 1958 con il titolo *The Computer and the Brain*, dove von Neumann analizza in profondità le differenze tra cervello e calcolatore digitale, prospettando (a parere di chi scrive) soluzioni che implicavano i prodromi per un superamento dei limiti della computazione digitale sequenziale. Un altro settore in cui il contributo di von Neumann fu assolutamente decisivo fu la teoria dei giochi. Già nel 1926 aveva elaborato la nozione del principio del *minimax*, pubblicata nel 1928 nell'articolo *Zur Theorie der Gesellschaftsspiele*, che getta le basi della teoria matematica dei giochi a due persone a somma zero. Questo sforzo culminò nel volume *Theory of Games and Economic Behavior* [von Neumann, Morgenstern, 1944], scritto con l'economista tedesco naturalizzato statunitense Oskar Morgenstern. Il testo fonda la teoria matematica dei giochi strategici, analizzando le decisioni razionali in contesti di conflitto e cooperazione, con ampie applicazioni all'economia, alla politica e alla strategia militare. John von Neumann si spense l'8 febbraio 1957 al Walter Reed Army Medical Center di Washington, all'età di 53 anni. Per una ricostruzione della vicenda umana e scientifica di John von Neumann si vedano Ulam, 1958; Heims, 1980; Aspray, 1990; Dyson, 2012 e Bhattacharya, 2021. La nota biografica di Stanislaw Ulam è particolarmente rilevante non solo per l'analisi del pensiero matematico, ma anche per la profonda vicinanza umana e professionale che legava l'autore a von Neumann. Ulam, emigrato negli Stati Uniti nel 1935 come *visiting professor* all'Università di Harvard, collaborò con von Neumann al progetto Manhattan a Los Alamos e, in seguito, alla formulazione della bomba all'idrogeno e agli studi sulla meccanica statistica. La sua testimonianza costituisce inoltre la prima ricostruzione bibliografica sistematica dell'opera scientifica di von Neumann. Ulteriori contributi sulla biografia e sul pensiero matematico del genio ungherese si trovano in Israel e Gasca (2008). Le *Opere complete* di John von Neumann sono raccolte nei cinque volumi curati da A. H. Taub, *Collected Works, 1903-1957* [1961], che coprono tutti gli svariati, segno di una curiosità intellettuale inappagabile, ambiti - dalla logica matematica alla fisica quantistica, dalla teoria dei giochi all'informatica teorica e applicata - toccati dal suo sterminato contributo.

²⁶² Per la storia dei calcolatori e delle tecnologie computazionali si veda Goldstine, 1972; Metropolis 1980; Ceruzzi, 2003; Ceruzzi, 2012; Haigh, Ceruzzi, 2021.

Non è stato invece così per Alan Turing. La sua lotta a Bletchley Park²⁶³, come figura chiave della crittoanalisi britannica impegnata contro il nemico nazista, i risultati raggiunti in quel contesto e l'influenza che il periodo bellico ha esercitato sul suo itinerario intellettuale e sulle scienze sviluppate anche grazie all'ispirazione del suo lavoro sono rimasti a lungo nell'ombra.

Prima di diventare noti all'utenza non specializzata – il suo lavoro a Bletchley Park iniziò a entrare nel dominio pubblico soltanto a metà anni '90 [Hodges, 2014, p. XIV, trad. it] – i contributi, le idee e la vicenda biografica di Turing rimasero per decenni poco conosciuti, non riconosciuti, in una parola: invisibili. Solo più tardi sarebbero stati valorizzati, grazie all'enfasi mediatica su *The Imitation Game*, al progredire delle lotte della comunità LGBTQ+ e dei movimenti antidiscriminazione che hanno visto in lui un simbolo, sino alla grazia postuma concessa da Elisabetta II la vigilia di Natale del 2013, rispetto alla condanna per “atti osceni” [*gross indecency*] [de Ceglia, 2014] inflittagli il 31 marzo 1952 e seguita dalla castrazione chimica mediante estrogeni, che anticipò di circa due anni la sua morte, avvenuta il 7 giugno 1954²⁶⁴.

²⁶³ Bletchley Park fu la sede della *Government Code and Cypher School* (GC&CS) durante le operazioni britanniche di intelligence nella Seconda Guerra Mondiale [Smith, 2006]. Situata a Milton Keynes, nel Buckinghamshire, a circa 80 km dal centro di Londra, la tenuta originaria si estendeva per oltre 581 acri e comprendeva terreni agricoli, boschivi e numerosi edifici ausiliari. Al centro della proprietà sorgeva la *mansion* padronale, circondata da 44 acri di parco paesaggistico, con giardini ornamentali, un lago, serre e un labirinto di tassi. La *mansion* fu completamente ampliata e rimodernata nel 1883 da Sir Herbert Leon, che l'aveva acquistata con la moglie nel 1882. L'edificio disponeva di 27 camere da letto, 7 bagni, 4 grandi saloni di ricevimento, una sala da ballo, garage per tre automobili, stalle per dodici cavalli, impianti elettrici e di riscaldamento autonomi, e alloggi per oltre 200 membri dello staff. Durante il conflitto, vi operarono alcune tra le figure più importanti della crittografia britannica, tra cui Alan Turing, Max Newman, Jack Good, Tommy Flowers, oltre a centinaia di matematici, linguisti, ingegneri e analisti. Fu qui che nacquero le macchine Bombe e il calcolatore elettronico Colossus, strumenti chiave nella decifrazione dei codici tedeschi. Secondo le stime, oltre 10.000 persone lavorarono a Bletchley Park nel periodo bellico: circa un terzo erano civili, e di questi il 75% erano donne. Queste persone furono soprannominate “geese”, oche, dal giudizio di Winston Churchill che le definì “*the geese who laid the golden egg and never cackled*”, ovvero “le oche che deponevano uova d'oro e non starnazzavano mai”, in riferimento al preziosissimo contributo strategico e alla straordinaria riservatezza che seppero mantenere. Su Bletchley Park e le “geese” e le loro condizioni di vita e di lavoro nel servizio di crittoanalisi britannico, si veda Hill, 2004.

²⁶⁴ Alcuni tra i maggiori biografi di Turing sono scettici intorno allo stabilire una correlazione chiara tra gli effetti (fisici e psicologici) della castrazione chimica e il suicidio del matematico inglese. Scrive Copeland: “Se cerchiamo di scoprire un motivo per il suicidio, è sufficiente considerare la «castrazione chimica»? D'altra parte Turing aveva sopportato il processo e la mortificante «terapia» chimica che ne conseguì con quella che Hilton descrisse come «spensierata forza d'animo». Le dosi ormonali erano cessate nell'aprile del 1953, più di un anno prima della sua morte. Dopo dodici mesi di «organoterapia» inumana, Turing continuava a trovare «sensuali» (utilizzò questa parola in una lettera a Robin Gandy) i giovani uomini che conosceva. All'inizio dell'estate del 1953 preparò la valigia per una vacanza al nuovo Club Méditerranée a Ipsos, sull'isola greca di Corfù. Sole, spiagge, uomini. Se Turing stava soffrendo di una depressione post trattamento che poteva mettere a rischio la sua vita, non ce n'è traccia. Inoltre la carriera era in uno dei suoi momenti migliori, con le ricerche sulla crescita che procedevano alla grande e con la prospettiva di nuovi risultati di importanza basilare proprio dietro l'angolo” [Copeland, 2012, p.306 trad.it.]. Hodges esprime una opinione simile. “La morte di Alan Turing fu un brutto colpo per quelli che lo conoscevano, perché non rientrava in nessuna chiara e plausibile concatenazione di eventi. In essa nulla era esplicito. Non c'erano stati segni premonitori, non fu lasciata spiegazione alcuna” [...] Che Alan fosse una persona infelice e in stato di continua tensione; che fosse in cura da uno psichiatra; che avesse sofferto un colpo capace di

Per ironia della storia, è proprio il carattere dell'*invisibilità*, della *dematerializzazione* - e dell'*ubiquità* - si pensi agli sviluppi più recenti dell'*Internet of Things* (IoT), intesa come miliardi di dispositivi interconnessi tra loro, che estenderanno i confini del mondo integrando entità fisiche e componenti virtuali [Li et al., 2015], oppure alle forme sempre più avanzate di interazione umana con sistemi di AI generativa, che ripresentano la tendenza strutturale a inglobare nella nostra architettura cognitiva “estesa” risorse non biologiche [Clark, 2025] - a costituire il tratto distintivo dei nuovi ambienti tecnologici sempre più avvolti [*enveloped*] dalle tecnologie. Ambienti in cui i nostri contesti di vita, grazie alla crescente potenza computazionale e allo sviluppo di sistemi statistici sempre più raffinati, sono sempre più adatti all'IA, progettati per essere compatibili con le tecnologie digitali [Floridi, 2022, pp. 53-56, 332, trad. it.].

Ciò pone la questione decisiva del design di queste tecnologie e dei modi e del *rapporto/interfaccia/interazione* dell'essere umano con essa e con le sue conseguenze [Floridi, 2014, pp. 37-42, trad. it.; Floridi, 2022, pp. 56-58, trad. it.].

Una traiettoria scientifica e tecnica la cui scintilla fu accesa, tra gli altri, dalle ricerche di Turing, eroe eponimo dell'era della *Quarta Rivoluzione*: l'epoca in cui l'essere umano perde il monopolio sull'azione intelligente, sul ragionamento logico e sulla capacità di processare informazioni [Floridi, 2014, p. 105].

Shannon aveva mostrato come ogni forma di comunicazione – dalla crittografia alla trasmissione dei geni del DNA – potesse essere trattata in termini discreti e stocastici. Parallelamente, la cibernetica, sin dai suoi esordi, aveva assunto che il feedback, struttura fondamentale di ogni sistema – organico o inorganico, naturale o artificiale – dovesse essere compreso in relazione alla “più fondamentale nozione di messaggio, fosse questo trasmesso per via elettrica, meccanica o nervosa”, intendendo per messaggio “una sequenza discreta o continua di eventi misurabili distribuiti nel tempo; precisamente ciò che gli statistici chiamano una serie temporale” [Wiener, 1948, pp. 32-33, trad. it.]. Possiamo collocare accanto a questi risultati il lavoro di Turing, il quale, a partire dai suoi studi sulla computabilità, può essere

abbattere molti uomini, tutto questo era noto e chiaro a tutti. Ma dal processo erano passati due anni; il trattamento a base di ormoni era finito un anno prima; Alan sembrava esserne ormai uscito del tutto”. Tuttavia, prosegue il biografo del matematico inglese, “chi non riusciva a collegare in maniera stereotipata il processo del 1952 con la morte del 1954 dimenticava forse che il suicidio non va necessariamente interpretato in termini di debolezza o vergogna” [Hodges, 1985, pp.633-634 trad. it.]. L'inchiesta il 10 giugno 1954 stabilì ad ogni modo che si trattò di suicidio: Turing mangiò una mela avvelenata. Una terza mela “mitologica” della storia della scienza, dopo quella mangiata da Eva e quella che si narra cadde sulla testa di Newton.

considerato una riflessione sui *limiti* della capacità computazionale della mente e sulla possibilità delle macchine di riprodurre le “facoltà” del cervello.

È peculiare come la dicotomia concettuale tra visibilità e invisibilità abbia attraversato le analisi delle pagine precedenti: da un lato la riduzione della matematica a puro gioco sintattico e dell'intelligenza a sistema di manipolazione simbolica trasparente, dall'altro il “residuo” che Turing ha tentato di riprodurre attraverso le sue macchine non organizzate.

Come vengono codificati questi messaggi? Nella metafora cibernetica che abbiamo analizzato, attraverso un sistema binario a due stati - 0 o 1, aperto o chiuso, vero o falso, acceso o spento - funzionalmente indistinguibile indipendentemente dal fatto che sia implementato da un relè o da un neurone:

È degno di nota il fatto che i sistemi nervosi umano e animale, che com'è noto sono in grado di svolgere il lavoro di un sistema di calcolo, contengano elementi idealmente capaci di agire come relé. Tali elementi sono i cosiddetti neuroni o cellule nervose. Mentre sotto l'influsso di correnti elettriche essi mostrano proprietà piuttosto complesse, la loro normale attività fisiologica è strettamente conforme al principio del «tutto o niente» [...] Il nervo può quindi essere considerato un relé essenzialmente a due stati: scatto e riposo [Wiener, 1948, pp. 163-164 trad. it].

Shannon, nel trattamento algebrico dei relè, aveva mostrato come le leggi di Boole - le “leggi del pensiero” - potessero essere realizzate da semplici circuiti elettrici, gli stessi a cui da giovane laureando magistrale doveva badare all'interno della scatola di controllo nascosta dentro l'enorme massa metallica dell'Analizzatore Differenziale di Bush. McCulloch e Pitts proposero un modello astratto in cui il neurone veniva formalmente assimilato a un circuito logico elementare, suggerendo che reti di neuroni, almeno in linea di principio, potessero computare qualsiasi funzione logica booleana.

Ricorda Wiener a proposito del nesso/*blurring* tra dimensione elettromeccanica (e poi elettronica), logica e neurologica che andiamo delineando:

A quel tempo Pitts aveva già una profonda preparazione in campo logico-matematico e neurofisiologico, ma non aveva avuto molte occasioni di venire a contatto con l'ingegneria. In particolare, non era al corrente del lavoro di Shannon e non aveva avuto molte esperienze in campo elettronico. Fu molto interessato quando gli mostrai esemplari di moderne valvole elettroniche e gli spiegai come queste fossero mezzi ideali per realizzare tecnicamente l'equivalente dei suoi circuiti e sistemi neuronici. Sin da quel tempo ci apparve evidente che il calcolatore ultrarapido, basato com'è su

dispositivi di commutazione consecutiva, doveva rappresentare un modello quasi ideale dei problemi che nascono dal sistema nervoso. Il carattere di *tutto o niente* della scarica dei neuroni corrisponde perfettamente alla scelta singola fatta nella determinazione di una cifra nella scala binaria, che più d'uno fra noi aveva già preso in considerazione come la più soddisfacente base di progettazione del calcolatore. La sinapsi altro non è se non un meccanismo per determinare se una certa combinazione di uscite da altri elementi selezionati agirà o no come stimolo adeguato a produrre la scarica dell'elemento successivo, e deve avere il suo esatto corrispondente nel calcolatore. Il problema di interpretare la natura e le varietà della memoria animale ha il suo equivalente nel problema di costruire memorie artificiali per la macchina [Wiener, 1948, pp.37-38 trad. it.].

Il cervello trova così, in consonanza con una certa interpretazione delle idee nate con *On Computable Numbers* [Turing, 1938], un suo omologo – o meglio, un modello rappresentativo ideale – nel calcolatore. Gli elementi, o 'organi', come li definirà von Neumann nel delineare l'architettura dell'EDVAC – in un'identificazione che segnerà un progresso decisivo nell'evoluzione dei calcolatori – rinviano immediatamente al problema della memoria e dell'implementazione delle funzioni logiche: dal 'nastro infinito' della macchina di Turing, alle prime forme di memoria elettronica e magnetica, sino all'integrazione nei circuiti logici del microprocessore, sviluppi che si riveleranno cruciali per la storia del computer e per il suo rapporto con l'intelligenza umana. In entrambi i casi, infatti, viene realizzata la medesima funzione computazionale. Ciò implica che, nella misura in cui il sistema nervoso può essere considerato come una delle possibili implementazioni della logica binaria, esso sia soggetto agli stessi limiti di qualsiasi meccanismo di calcolo. Da questo punto di vista, i confini della Tesi di Church-Turing si applicano anche alle possibilità computazionali dei neuroni.

Ne offre una testimonianza lo stesso Warren McCulloch. Se in *A Logical Calculus* [1943] si era posta - sebbene non provata - l'equivalenza tra rete, intesa come modello neuronale semplificato, e macchina di Turing e il nome del matematico inglese veniva citato due volte rispetto alla capacità di calcolo del neurone²⁶⁵, in quello che rappresenta "il primo

²⁶⁵ McCulloch e Pitts non fornirono mai una dimostrazione che il loro modello di rete neurale fosse Turing-equivalente. In *A Logical Calculus*, si limitarono ad affermare che vi erano risultati da dimostrare, ma non presentarono alcuna prova formale della potenza computazionale delle loro reti [Piccinini, 2004]. L'unico punto del loro celebre articolo in cui viene citato esplicitamente Turing è quello in cui i due autori affermano chiaramente che una rete è computazionalmente equivalente a una macchina di Turing solo se è dotata di un nastro, di scanner connessi agli afferenti (cioè all'ingresso del sistema), e di effettori adatti a compiere le operazioni motorie necessarie. In caso contrario - cioè in assenza di tali strumenti esterni - la rete può calcolare solo alcuni, e non tutti, i numeri calcolabili da una macchina di Turing: "One more thing is to be remarked in conclusion. It is easily shown: first, that every net, if furnished with a tape, scanners connected to afferents, and suitable efferents to perform the necessary motor-operations, can compute only such numbers as can a Turing machine; second, that each of the latter numbers can be computed by such a net; and that nets with circles can compute, without scanners and a

collegamento pubblicato noto tra la teoria matematica della computazione e la teoria del cervello” [Piccinini, 2004 trad. it. mia]²⁶⁶ McCulloch riconobbe esplicitamente il ruolo delle teorie di Turing nella direzione generale della sua ricerca, durante il dibattito che seguì la presentazione di von Neumann intitolata *The General and Logical Theory of Automata* [von Neumann, 1951], incentrata sulla possibilità di formalizzazione degli automi, sul rapporto tra macchine computazionali e sistema nervoso, e sulla realizzabilità di sistemi artificiali auto-replicanti. L'intervento fu presentato nel corso della conferenza interdisciplinare *Cerebral Mechanisms in Behavior*, tenutasi nel settembre 1948 presso il California Institute of Technology e sponsorizzata dallo Hixon Fund - un evento che rappresenta una pietra miliare nella nascita delle scienze cognitive della computazione e della *computer science* [Gardner 1985, pp. 22-29, trad. it.; la trascrizione dello Hixon Symposium è in Jeffress, ed. 1951]. McCulloch ricordò come, nel suo itinerario che lo condusse dall'interesse iniziale per la filosofia, attraverso la matematica, fino alla neurofisiologia, nel tentativo di costruire solide basi epistemologiche per quest'ultima,

Ironia della sorte, iniziai completamente dalla parte sbagliata [...] solo quando vidi l'articolo di Turing cominciai ad avviarmi nella direzione giusta e, con l'aiuto di Pitts, formulammo il calcolo logico necessario [...] La cosa affascinante è che il più semplice insieme di assunzioni appropriate è sufficiente a dimostrare che un sistema nervoso può calcolare qualsiasi numero calcolabile. È quel tipo di dispositivo, se vuoi - una macchina di Turing [von Neumann, 1951, p.32-33 trad mia]²⁶⁷.

tape, some of the numbers the machine can, but no others, and not all of them” [McCulloch, Pitts, 1943]. Se si prende sul serio l'identificazione tra macchina di Turing, neurone e circuito di calcolo che il loro lavoro ha ispirato, si può dire che la semplice rete di McCulloch e Pitts, considerata nella forma priva di scanner, nastro ed effettori, corrisponde - rispetto alla struttura formale della macchina di Turing rappresentata dalla quintupla $\langle q_i, s_i \rightarrow s_i', D, q_i' \rangle$ - alla sola funzione di transizione, per di più vuota: senza input, senza simboli da scrivere o leggere, né direzioni di movimento. In termini di architettura computazionale, questo equivale ad avere soltanto il meccanismo di controllo logico della CPU, privo di accesso a memoria, canali di input/output o dati su cui operare. È tuttavia interessante come McCulloch e Pitts considerano questi limiti della rete. Viene interpretata come spiegazione psicologica della tesi di Church- Turing e cioè che i limiti computazionali dei loro modelli neurali corrispondano ai limiti computazionali dei cervelli biologici; e che questi, a loro volta, riflettano le limitazioni psicologiche effettive degli esseri umani impegnati in processi computazionali. Il processo di calcolo con carta e matita in cui è impegnato un essere umano sarebbe spiegato dalle reti di McCulloch-Pitts [Piccinini, 2004]. È stato Kleene, nel 1956, a risolvere le ambiguità del modello di McCulloch-Pitts e a stabilirne con rigore la portata computazionale, in un articolo comparso su *Automata Studies*, rivista curata da Claude Shannon e John McCarthy. Il matematico statunitense perfezionò il formalismo del modello delle reti, dimostrando che queste erano computazionalmente equivalenti ad automi a stati finiti (*finite automata*), cioè modelli computazionali costituiti da una rete di “celle” (*cells*), ciascuna delle quali può assumere un numero finito di stati interni distinti, e il cui comportamento dipende dallo stato corrente e dall'input ricevuto [Kleene, 1956].

²⁶⁶ “This brief passage is the only one mentioning computation. By stating that McCulloch-Pitts nets computed, this passage provided the first known published link between the mathematical theory of computation and brain theory. It was a pivotal statement in the history of computationalism”

²⁶⁷ “I'd like to say, historically, how I came to be interested in this particular problem, if you'll forgive me, because it does bear on this matter. I came, from a major interest in philosophy and mathematics, into psychology with the

Calcolatori, logica computazionale e sistema nervoso possono allora essere studiati in modo analogo, come le diverse facce di un unico prisma, in cui la differenza di campo dipende soltanto da una decisione analitica. Non sorprende, quindi, che Turing, nel suo *On Computable Numbers*, “parlasse (di fatto) già di un programma per computer: un insieme di regole definite dal software, ma implementabili in hardware” [Boden, 2006, p.176, trad. mia]. Sebbene si occupasse dei confini delle possibilità di formalizzazione simbolica, il lavoro di Turing del 1936 “introdusse il concetto di macchina nelle discussioni sui fondamenti della matematica” [Copeland, 2017, p.54, trad. mia]²⁶⁸.

Posta come fondamento delle scienze computazionali (e come modello della mente nelle scienze cognitive), la Macchina di Turing ha progressivamente assunto una sorta di dimensione platonica, di pura entità matematica, che ne accentuava il carattere idealizzato. Tuttavia, malgrado la descrizione astratta formulata in *On Computable Numbers*, “c’è un sapore decisamente industriale nel suo [di Turing] resoconto delle macchine, con i suoi riferimenti non solo al nastro di carta perforato, ma persino a ‘ruote’ e ‘leve’ all’interno dello scanner” [Copeland, 2017c, p.54, trad. mia]²⁶⁹.

La descrizione formale della Macchina di Turing - con le sue testine, i nastri cartacei, i riquadri su cui vengono stampati gli 0 e gli 1, i meccanismi di scorrimento - non va considerata solo dal punto di vista della matematica pura, ma come una rappresentazione teorica che racchiude in sé le potenzialità di essere implementata in una macchina fisica, quella di cui l’articolo del 1936 proponeva una trattazione teorica.

problem of how a thing like mathematics could ever arise-what sort of a thing it was. For that reason, I gradually shifted into psychology and thence, for the reason that I again and again failed to find the significant variables, I was forced into neurophysiology. The attempt to construct a theory in a field like this, so that it can be put to any verification, is tough. Humorously enough, I started entirely at the wrong angle, about 1919, trying to construct a logic for transitive verbs. That turned out to be as mean a problem as modal logic, and it was not until I saw Turing’s paper that I began to get going the right way around, and with Pitts’ help formulated the required logical calculus. What we thought we were doing (and I think we succeeded fairly well) was treating the brain as a Turing machine; that is, as a device which could perform the kind of functions which a brain must perform if it is only to go wrong and have a psychosis. The important thing was, for us, that we had to take a logic and subscript it for the time of the occurrence of a signal (which is, if you will, no more than a proposition on the move). This was needed in order to construct theory enough to be able to state how a nervous system could do anything. The delightful thing is that the very simplest set of appropriate assumptions is sufficient to show that a nervous system can compute any computable number. It is that kind of a device, if you like-a Turing machine”.

²⁶⁸ “Turing had brought machinery into discussions of the foundations of mathematics”.

²⁶⁹ “It seems transparently clear that Turing himself was a Turing-machine realist. here is a positively industrial flavour to his account of his machines, with its references not only to punched paper tape, but even to ‘wheels’ and ‘levers’ within the scanner”.

La realizzazione concreta in *hardware* della Macchina di Turing è culminata nel principio del programma memorizzato, lo “*stored program principle*” [Haigh, Ceruzzi, 2021]. Questo principio costituisce la base dell’architettura di ogni computer moderno, secondo cui “i computer memorizzano sia le istruzioni - cioè i programmi - sia i dati su cui tali istruzioni operano nello stesso dispositivo di memoria fisico, senza alcuna barriera fisica o progettuale tra di essi” [Ceruzzi, 2012, p. 29, trad. mia]²⁷⁰. Tale principio concretizza l’idea della macchina universale: una macchina di Turing specializzata può essere codificata numericamente e diventare essa stessa un input. “Il principio del programma memorizzato [...] cancellò la distinzione tra numeri che significano e numeri che fanno cose. Da allora il nostro universo non fu più lo stesso” [Dyson, 2012, p. IX, trad. it.].

Nelle pagine precedenti la nostra indagine è partita dalle contraddizioni e dai paradossi che nascono quando nei sistemi formali i simboli si rivolgono a sé stessi. La Macchina di Turing nasce per testare l’*Entscheidungsproblem* - stabilendo una sorta di linea di filiazione tra Via Galvani I a Bologna, sede dell’Aula Magna dell’Archiginnasio²⁷¹ dove Hilbert il 3 settembre 1928 tenne il suo discorso all’ottavo Congresso Internazionale dei Matematici e la San Francisco Bay Area - e quella Universale fece propria un’intuizione gödeliana, quella per cui non c’è differenza tra numeri, i dati, e operazioni su numeri, le istruzioni.

L’idea di Gödel era che le affermazioni logiche e aritmetiche potessero essere rappresentate tramite numeri. Turing, von Neumann e altri svilupparono questa idea molto più a fondo, ed essa è oggi onnipresente nell’informatica: ogni laptop contiene numeri binari che rappresentano istruzioni di programma, enunciati logici e aritmetici, e dati di ogni tipo [Copeland, 2017, p.60 trad. mia]²⁷².

L’incarnarsi di queste idee ha portato con sé - riproducendoli - anche quei limiti interni che, *in nuce*, contengono già i motivi di un possibile superamento della stessa struttura computazionale da esse originata. Si pensi, ad esempio, ai limiti della computabilità, che trovano un corrispettivo nelle facoltà proprie dell’architettura cognitiva umana; alla definizione

²⁷⁰ “Modern computers store both their instructions-the programs-and the data on which those instructions operate in same physical memory device, with no physical or design barrier between them”.

²⁷¹ Il programma dell’VIII Congresso Internazionale dei Matematici, tenutosi a Bologna nel 1928, è disponibile sul sito MacTutor History of Mathematics dell’Università di St Andrews alla pagina https://mathshistory.st-andrews.ac.uk/ICM/ICM_Bologna_1928/

²⁷² Gödel’s idea was that logical and arithmetical statements can be represented by numbers. Turing, von Neumann, and others developed this idea much further, and it is now ubiquitous in computing, with everyone’s laptop containing binary numbers that represent program instructions, logical and arithmetical statements, and data of all types.

ontologica di una nozione fondamentale come quella di algoritmo - che ha nel suo cuore una *simbiosi* che si instaura tra l'insieme di regole (la macchina) e l'essere umano; al legame con una descrizione dell'organizzazione funzionale del cervello.

Tutti questi elementi si inseriscono poi all'interno di una traiettoria culturale che ha condotto alla personificazione della computazione, lungo un asse che va "dallo strumento al partner" [*from tool to partner*] [Grundin, 2017].

La storia dell'*embodiment* delle idee che tracciano questo arco intellettuale, scientifico e tecnologico - che trova, paradossalmente, tra i suoi luoghi di nascita la scarnificazione di un essere umano impegnato in un atto di calcolo, e prima ancora l'astrazione di ogni caratteristica particolare, corporea, materiale, concreta (si ricordi il celebre detto hilbertiano: «si deve poter dire in ogni momento - invece di punti, rette e piani - tavoli, sedie e boccali di birra») - può essere ricostruita attorno ad alcuni assi concettuali, considerati nella loro interconnessione reciproca.

Questi assi sono quelli individuati dallo storico della scienza e della tecnologia informatica nonché *curator emeritus* dello *Smithsonian's National Air and Space Museum* di Washington, D.C., Paul Ceruzzi: il paradigma digitale, la convergenza tecnologica, lo sviluppo dell'elettronica a stato solido e l'evoluzione del rapporto tra uomo e macchina [Ceruzzi, 2012, pp. X-XIV].

- *il paradigma digitale*: l'idea di codificare l'informazione, le istruzioni, i dati e le attività computazionali, logiche di controllo logico in forma binaria, usando due simboli 0 e 1 - come quelli stampati dalla macchina di Turing sul nastro, i due stati aperto/chiuso dei circuiti e i due valori booleani Vero e Falso - al posto dell'aritmetica in base dieci.

- *la convergenza tecnologica*: nello sviluppo delle tecnologie informatiche, un ruolo fondamentale è svolto dalla convergenza di traiettorie tecnologiche differenti, inizialmente indipendenti tra loro, che vengono via via messe in relazione e integrate grazie al comune paradigma digitale. L'esempio più emblematico di questo processo è rappresentato dall'odierno smartphone, dispositivo che incarna l'unione di molteplici tecnologie eterogenee

- *l'elettronica a stato solido*: dai relè ai microchip, passando per i transistor, la storia dello sviluppo computazionale e informatico è caratterizzata da un miglioramento continuo - in un'accelerazione all'apparenza continua e progressiva (scandita dalla "legge di Moore") - della tecnologia di base.

- *L'interfaccia uomo-macchina*: le modalità con cui l'essere umano si interfaccia ed entra in relazione con le tecnologie digitali. Queste ruotano intorno a una domanda fondamentale:

“Stiamo cercando di creare un sostituto meccanico dell’essere umano, o uno strumento che lavori in simbiosi con l’uomo, un’estensione delle sue facoltà mentali?” [Ceruzzi, 2012, p.42, trad. mia]²⁷³. E, nel secondo caso, in che modo entra in gioco la natura dell’architettura neuro-cognitiva umana, intesa nel senso delle scienze che l’hanno interpretata come elaboratore di informazione?

Qual è il design - dell’hardware e del software - che devono avere queste macchine che entrano in rapporto con l’essere umano, e che ruolo hanno nell’integrarne - o amplificarne - le capacità? E come entrano in gioco, da un lato, la dimensione corporea umana - abbiamo visto, all’inizio del primo percorso, il ruolo decisivo dell’*embedding* della dimensione cinestetica del pilota/bombardiere nella progettazione del predittore antiaereo, luogo genetico della cibernetica - e, dall’altro, quella cognitiva, nel gestire quantità sempre maggiori, nel senso che abbiamo visto in Shannon, di informazione?

Questo in un orizzonte in cui - come nel caso degli LLMs o dei linguaggi di programmazione sempre più di alto livello - si riesce a interagire con le macchine in linguaggio naturale, e queste diventano sempre più parte dell’ecologia cognitiva distribuita umana.

Secondo queste strade, come vedremo nei capitoli successivi, si è delineato lo sviluppo del calcolatore, dalle macchine di Bletchley Park, che materializzavano i metodi statistici di risoluzione crittografica, sino al “paradigma EDVAC”, l’archetipo del calcolatore a programma memorizzato costruito secondo un’architettura seriale e che si strutturava, e riproduceva, anche a livello di impatto sociale, la “metafora cibernetica” sin qui incontrata.

È ora utile a tal fine analizzare un tentativo che costituisce un misconosciuto antecedente storico *in hardware* dell’implementazione di alcuni concetti – dal paradigma numerico, all’uso dei *vacuum tubes* e di tecnologie di scansione – che conteneva già in nuce importanti conseguenze per il design e la logica del calcolatore. Protagonisti di questa vicenda furono due studiosi già incontrati, il cui ruolo fondamentale è stato più volte sottolineato: Norbert Wiener e Vannevar Bush. Lo scopo era quello di affrontare in modo appropriato un formidabile avversario di ogni computer, umano o artificiale: le equazioni differenziali

2. Bush e Wiener a caccia all’equazione differenziale

²⁷³ “Do we construct machines that do what is technically feasible and adapt the human to their capabilities, or do we consider what humans cannot do well and try to construct machines that address those deficiencies?”

Alla fine del 1940, come abbiamo visto in precedenza, Wiener testò le equazioni della sua teoria matematica – quelle contenute nello *Yellow Peril* – alla base dell'Antiaircraft Predictor, utilizzando l'Analizzatore Differenziale di Vannevar Bush. Il macrocontesto bellico in cui il dispositivo wieneriano si inseriva, in un peculiare ritorno al punto di partenza del percorso che stiamo delineando, condusse al superamento del paradigma computazionale rappresentato dal *Differential Analyzer* e all'affermazione del computer moderno, secondo i fattori di sviluppo appena discussi. Considerare la relazione intellettuale tra il fondatore della cibernetica e il futuro capo dell'NDRC può risultare utile per mettere in prospettiva alcuni dei concetti in gioco.

L'inizio dei contatti tra Norbert Wiener e Vannevar Bush risale alla seconda metà degli anni '20, al MIT - centro che, grazie all'influenza di Bush, sarebbe divenuto uno dei poli principali nello sviluppo dei calcolatori [Goldstine, 1972, p. 51]²⁷⁴. Si trattava di un periodo in cui, all'interno dell'istituto del Massachusetts, vigeva una collaborazione tra scienziati animata da un forte *esprit de corps* [Birkhoff, 1980, pp. 30-31]²⁷⁵.

Bush, allora impegnato nella costruzione della sua macchina calcolatrice fatta di ingranaggi, dischi e alberi meccanici d'acciaio per la risoluzione delle equazioni differenziali chiese l'aiuto di Wiener. Questi, interessato sin dall'infanzia alla costruzione di automi e in quel periodo profondamente immerso nello studio dell'analisi armonica, accettò la proposta, poco prima di partire come *Guggenheim Fellow* per Göttingen, la "casa" del grande Hilbert e, prima dell'ascesa del nazismo, capitale mondiale della matematica Hilbert. [Masani, 1990, p. 105; Conway, Siegelman, 2005, pp.84-91 trad. it.]

La macchina di Bush non riusciva a calcolare le Equazioni Differenziali Parziali (PDE), "mostri della matematica in cui due o più fattori mutavano simultaneamente, una situazione che si presentava spesso nella tecnologia delle correnti deboli" [Conway, Siegelman, 2005, p.107]: La risoluzione delle PDE era necessaria per dare all'*Heaviside Operational Calculus*, usato nella progettazione dei circuiti una rigorosa fondazione logico-matematica²⁷⁶.

²⁷⁴ "It was Bush's great influence, which was to make MIT one of the greatest centers for innovation in the field of computing".

²⁷⁵ "Bush communicated frequently with Wiener-there was a lot of *esprit de corps* in the small MIT of those days".

²⁷⁶ Il calcolo operativo, "oscuro" e "idiosincratico" [Conway, Siegelman, 2005], formalizzato dal fisico e matematico britannico Oliver Heaviside (1850-1925), è stato per lungo tempo l'unico corpus metodologico a disposizione degli ingegneri che stavano sviluppando le tecnologie telegrafiche e telefoniche, tra la fine dell'800 e l'inizio del '900. Il lavoro di Heaviside [Heaviside, 1893], seppur "brillante", era "puramente euristico, privo persino del minimo intento di rigore matematico" [The brilliant work of Heaviside is purely heuristic, devoid of even the pretense to mathematical rigor. Wiener, 1926, p.558]. Tuttavia era utile nella pratica: "Although Heaviside's developments have not been justified by the present state of the purely mathematical theory of operators, there is a great deal of what we may call experimental evidence of their validity, and they are very

L'idea di Wiener fu quella, balenatagli in mente mentre assisteva ad uno spettacolo al Copley Theatre di Boston, di un calcolatore *ottico*.

Stavo guardando lo spettacolo al Copley Theatre [e] mi venne un'idea che semplicemente distraesse la mia attenzione dall'esibizione. Si trattava di una macchina calcolatrice ottica per l'analisi armonica. Avevo già imparato a non ignorare quelle idee fortuite, a prescindere da quando mi balzavano in testa, e subito uscii dal teatro per elaborare alcuni dettagli del mio nuovo progetto. Il giorno successivo mi consultai con Bush [Wiener, 1956, p.112 trad. it. in Conway, Siegelman, 2005, pp.107-108].

Wiener - che fa un resoconto di questa idea nelle pagine introduttive di *Cybernetics* - voleva utilizzare la scansione ottica, che stava prendendo corpo nella nascente tecnologia televisiva, come strumento computazionale: "Ero anche convinto che la soluzione del problema risiedeva nel procedimento di scansione, come viene impiegato in televisione, e infatti la televisione era destinata a essere più utile all'ingegneria per l'introduzione di questa nuova tecnica che non come industria indipendente" [Wiener, 1948, p. 27 trad. it.].

Per realizzare questa macchina, Wiener propose a Bush - con un'intuizione gravida di futuro, ma in anticipo sui tempi di un ventennio - di avvalersi della tecnologia in rapida evoluzione, quella del tubo a vuoto [Conway, Siegelman, 2005, p.107]. Il calcolatore che aveva in mente doveva basarsi "su tubi elettronici piuttosto che su ruotismi o relé meccanici, al fine di assicurare un'azione più rapida", in contrasto con la tragica lentezza della macchina di Bush nel risolvere il numero enorme di calcoli necessari per un'equazione differenziale parziale. Inoltre, era necessario che "l'apparato centrale di somma e moltiplicazione della macchina calcolatrice fosse numerico" [Wiener, 1948, p. 27 trad. it.].

Queste idee avveniristiche, tuttavia troppo in anticipo sullo sviluppo tecnologico affinché potessero realizzarsi, non attecchirono. Tuttavia la collaborazione tra Wiener e Bush

valuable to the electrical engineer" [Wiener, 1926, p.558]. "I metodi di Heaviside funzionavano per magia nera: le risposte che fornivano erano corrette, ma nessuno sapeva perché" [Raisbeck, comunicazione personale 23 luglio 1999, in Conway, Siegelman 2005, p.96 n.8 trad. it.]. Data la mancanza di rigore del calcolo di Heaviside, "il MIT [tramite Bush] si rivolse a Wiener perché ponesse fondamenta solide alla base delle 'nuove e potenti tecniche di comunicazione' di Heaviside" [Conway, Siegelman, 2005, p.77 trad. it.]. Heaviside fu protagonista nel 1900 di "uno degli episodi più discussi negli annali degli albori della tecnologia della comunicazione". La "bobina a carico" - dispositivo di alimentazione che rese possibile le telefonate interurbane - inventata, ma mai brevettata, dal fisico inglese nel 1887, fu veduta invece brevettata dal professore di ingegneria Michael Pupin alla AT&T - con la probabile complicità di quest'ultima - per l'astronomica, ai tempi, cifra di 750.000 dollari. "Pupin guadagnò fama, grandi fortune e un premio Pulitzer per la saga che scrisse sul suo passaggio dalle stalle alle stelle; mentre lo sventurato Heaviside morì in miseria, amareggiato e mezzo pazzo in un remoto cottage balneare inglese nel 1925" [Conway, Siegelman 2005, pp.430-431 trad. it.].

non si interrompe. Bush, nel 1929, lavorò a un testo, rivolto a ingegneri e studenti di ingegneria, *Operational Circuit Analysis* [Bush, 1929], in cui espose in maniera sistematica - con una formalizzazione simbolica e strumenti di analisi matematica - il calcolo operativo introdotto da Heaviside, per il trattamento dei circuiti elettrici, meccanici e termici. Wiener scrisse l'appendice del testo, intitolata *Fourier Analysis and Asymptotic Series* [Wiener, 1929], in cui fornì una elegante e rigorosa base matematica al metodo applicato ai circuiti e le fondamenta degli operatori simbolici applicati da Bush attraverso gli strumenti dell'analisi di Fourier. Un giudizio della collaborazione lo dà lo stesso Bush nella prefazione del testo: "Non sapevo che un ingegnere e un matematico potessero divertirsi così tanto insieme. Vorrei solo riuscire ad afferrare in modo autentico e profondo la logica fondamentale della matematica, come lui afferra i principi basilari della fisica" [Bush, 1929, pp.V-VI trad. mia]²⁷⁷. Anche Wiener conservò ricordi piacevoli di questo lavoro comune, che prometteva future collaborazioni [Wiener, 1956b, p. 139]²⁷⁸, quasi un segno della futura convergenza disciplinare e tecnologica che caratterizzerà lo sviluppo dei calcolatori e delle scienze ad essi collegate.

Il concetto di *scansione* ritornò nel dialogo tra Wiener e Bush. In una lettera del 20 settembre del 1940, Wiener, che cercava una maniera di dare concretezza alla sua "personale sfida bellica" - poi culminata con il predittore antiaereo - contro il nazifascismo, propose all'allora capo dell'NDRC un sistema computazionale automatico di risoluzione delle PDE basato proprio su questo principio e ribadito con forza: "Ripeto: il contributo principale che credo di offrire al problema della soluzione meccanica delle equazioni differenziali alle derivate parziali è l'uso sistematico dell'idea di scansione. Questo è economicamente importante, poiché consente di utilizzare apparati costosi in modo ripetuto ed efficiente" [Wiener, lettera a V. Bush, 20 settembre 1940, trad. mia]²⁷⁹. Tuttavia, rispetto alla avveniristica scansione ottica proposta a Bush negli anni '20, lo *scanning* teorizzato da Wiener era di natura elettromeccanica, basata su un nastro metallico [*steel tape*] usato come supporto magnetico per la registrazione e la scansione dei dati [Wiener, lettera a V. Bush, 20 settembre 1940]²⁸⁰. Questo sistema, che rendeva *discreti* i dati in una griglia [Masani, 1987, p.184], avrebbe permesso, secondo Wiener,

²⁷⁷ "I did not know an engineer and a mathematician could have such good times together. I only wish that I could get the real vital grasp of the basic logic of mathematics that he has of the basic principles of physics".

²⁷⁸ "We enjoyed the collaboration greatly, and both of us have spoken often of the fun we had in working together".

²⁷⁹ "I repeat the main contribution which I have bare to make to the problem of the mechanical solution of partial differential equations is the consistent use of the idea of scannings; this is economically important as it allows expensive apparatus to be used repeatedly and efficiently".

²⁸⁰ "That is, this tape will contain ten parallel lines marked and scanned magnetically, and these lines will indicate a number between 0 and 1024 according In this replacement the boundary of the net is left untouched".

con poco dispendio economico, la risoluzione di una equazione differenziale come quella di Laplace in circa 3,89 ore con un margine di errore dello 0,1% [Wiener, lettera a V. Bush, 20 settembre 1940 trad. mia]²⁸¹.

Il meccanismo ideato da Wiener per risolvere le equazioni differenziali in modo più rapido, affidabile e con minore spreco di risorse introduceva un ulteriore elemento che, come vedremo, si rivelerà decisivo: il sistema di computazione binario. “Propongo che i valori siano registrati in sistema binario su un nastro d'acciaio senza fine, nel consueto modo magnetico” [Wiener, lettera a V. Bush, 20 settembre 1940 trad. mia]²⁸².

La lettera di Wiener a Bush accompagnava un rapporto dattiloscritto - parzialmente illeggibile e costellato di errori, corretto e integrato da Pesi Masani, curatore delle opere complete commentate di Wiener e inserito nel volume IV dei *Norbert Wiener: Collected Works* - intitolato *Memorandum on the Mechanical Solution of Partial Differential Equations* [Wiener, 1940]. In questa nota, approfondendo gli aspetti matematici e tecnici già delineati nella lettera, Wiener arriva a descrivere le fattezze di un calcolatore elettronico ad alta velocità finalizzato a un compito specifico [Ferry, Saeks, 1987, p. 194]²⁸³.

Per la risoluzione ad alta velocità delle equazioni differenziali proposte, Wiener ribadisce l'adozione della computazione *binaria*, associata a un meccanismo rapido per imprimere dati su un nastro, insieme a un altro che ne legga i valori codificati come 0 e 1 [Wiener, 1940 trad. mia]²⁸⁴. Tra le varie soluzioni prese in considerazione per sommare i valori letti dal meccanismo e re-imprimerli su nastro, egli ritorna su un'innovazione tecnologica che, come vedremo, rappresenterà un passo decisivo verso il computer moderno: il *tubo a vuoto*, capace di incrementare la velocità computazionale di svariati ordini di grandezza. “Mi è stato

²⁸¹ “I shall take up as a particular example a method for solving the boundary value problem for the Laplace equation [...] This partial difference equation is solved by an apparatus which repeatedly scans a collection of data recorded on some very inexpensive device and replaces these data by new data [...] If we are aiming at an accuracy of one part in 1000, this means that 1.4 scanings are necessary and therefore that on each line 140,000,000 impulses must be recorded before the scanning is complete. Assuming a rate of scanning of 10,000 impulses per second, this yields a time not exceeding 3.89 hours for the final answer to be obtained”

²⁸² “I propose that the values be recorded in the binary system on an endless steel tape in the usual magnetic manner”.

²⁸³ “This September 1940 letter to Vannevar Bush and attached memorandum represents a tour de force in the computing art of the day. Indeed, Wiener synthesizes the best ideas of the time, together with some new innovations of his own, into a proposal to build a digital partial difference analyzer that exhibits every characteristic of the modern digital computer except for the stored program”.

²⁸⁴ “A. A quick mechanism for imprinting numerical values on a running tape. I suggest that these values be carried in the binary scale, as a number of lines on which a signal may be turned on or off, to represent a digit 1 or 0 in the corresponding place in the binary scale. [...] B. B. A mechanism for reading four such imprinted values simultaneously at fixed stations on the tape as it moves by”.

detto che esistono meccanismi a tubi a vuoto e capacità di questo tipo” in grado di operare “con una velocità complessiva di 1/50.000 di secondo per operazione” [Wiener, 1940 trad. mia]²⁸⁵.

La risoluzione delle PDE implica un numero talmente elevato di operazioni che “il vantaggio economico delle macchine aritmetiche elettroniche ad alta velocità, combinate con processi di scansione, rispetto alla moltiplicazione di parti meccaniche diventa così grande da renderne l’uso imperativo” [Wiener, 1940 trad. mia]²⁸⁶. I dispositivi di calcolo delineati da Wiener avrebbero dovuto combinare “macchine aritmetiche elettroniche ad alta velocità” con “dispositivi altrettanto rapidi per l’archiviazione e la riproduzione dei dati” [Wiener, 1940 trad. mia]²⁸⁷.

Il calcolatore teorizzato da Wiener, nelle sue caratteristiche principali - algoritmo discreto per la risoluzione delle PDE, aritmetica binaria, data storage, nastro magnetico multitraccia e unità logico-aritmetica elettronica - si pone allora nel solco tracciato dalla macchina di Turing e dalle sue incarnazioni successive: “Dal punto di vista dell’architettura, l’analizzatore alle differenze parziali proposto da Wiener è un classico esempio da manuale di macchina di Turing. Wiener non ha lasciato alcuna traccia di un incontro prebellico con Turing, sebbene debba certamente aver conosciuto il celebre articolo di Turing del 1936” [Ferry, Saeks, 1980, p. 195, trad. mia]²⁸⁸.

La proposta di Wiener, tuttavia, differisce da quella classica dell’architettura di von Neumann: si tratta di una macchina *special purpose*, finalizzata a un compito specifico, che tuttavia prefigura già un’articolazione funzionale assimilabile all’idea di ‘software’, intesa come sistema di controllo logico delle operazioni. Nel progetto wieneriano, ciò corrisponde all’insieme coordinato di dispositivi per lo *scanning*, la lettura e la stampa magnetica dei dati su cui la macchina esegue le operazioni. Wiener suggerisce una sorta di ipotesi funzionalista quando afferma che “il nastro d’acciaio e altri dispositivi simili sono quasi indipendenti dalla

²⁸⁵ “A rapid adding mechanism for adding the values given in B to a single number for reimprinting on the tape as in A. I am told that vacuum tube-capacitance mechanisms of this type exist with an overall speed of 1/50,000 sec. per operation”.

²⁸⁶ “Where operations are so multiplied in number as is the case with partial differential equations as distinguished from ordinary differential equations, the economic advantage of high-speed electronic arithmetical machines, combined with scanning processes, over the multiplication of mechanical parts, becomes so great that it is imperative”.

²⁸⁷ “Every such apparatus must combine with the high speed arithmetical machines adequate high-speed devices for the storage and reproductions of data”.

²⁸⁸ “Architecturally Wiener’s proposed partial difference analyzer is a textbook example of a Turing machine. Wiener has left no record of any prewar meeting with Turing, though he certainly must have known of Turing’s classic 1936 paper”.

natura specifica del problema da risolvere” [Wiener, 1940 trad. mia]²⁸⁹. Ciò implica una certa indipendenza del supporto materiale rispetto alle operazioni da eseguire, ma allo stesso tempo riconosce che l’implementazione concreta delle regole è condizionata dalla configurazione fisica del dispositivo. Le regole di funzionamento dovevano realizzarsi in *hardware*. Wiener “auspicava che il programma venisse «compreso nella macchina stessa»” [Conway, Siegelman, p.155 n.5 trad. it.].

Questa concezione - che intreccia dispositivo fisico, e regole operative riferite a un compito - esprime un nucleo profondo del pensiero di Wiener e della cibernetica, ovvero il legame strutturale tra computazione e dimensione “somatotopica” [Arbib, 1972]. Vedremo, più avanti come nel pensiero dello stesso von Neumann, l’introduzione nel cuore stesso della computazione della teoria dell’informazione, l’approfondimento della metafora cibernetica porti a una modifica del design della macchina, e della sua logica in senso neuromorfico.

Da questo punto di vista, una genealogia del paradigma connessionista e del calcolo in parallelo può essere ricostruita seguendo le tracce fino al *Memorandum* di Wiener del 1940: nella descrizione dell’hardware, egli propone quella che è forse la più “sorprendente e lungimirante” delle sue intuizioni, ovvero il nastro magnetico multitraccia. “La realizzazione da parte di Wiener del ‘nastro’ di Turing come un nastro magnetico multitraccia, in cui i ‘bit’ di un numero binario sono registrati simultaneamente su tracce parallele, anticipò l’industria di almeno un quarto di secolo” [Ferry, Saeks, 1987, p. 196 trad. mia]²⁹⁰.

Il calcolatore ipotizzato da Wiener esprimeva una concezione presente in tutto il suo pensiero riguardo l’uso delle macchine. Un’idea che risponde a una delle domande fondamentali riguardo all’interazione tra uomo e tecnologia. Per Wiener, “il calcolatore elettronico era un ausilio protesico per permettere alla mente umana di trascendere i propri limiti e svolgere meglio i suoi compiti” [Masani, 1987, p. 184 trad. mia]²⁹¹.

Le PDE erano dei compiti di calcolo soverchianti. Ma l’uso del calcolatore poteva essere esteso ad altri compiti “nei quali l’attuale teoria risulta computazionalmente così complessa da essere quasi inutilizzabile”. Compiti come “tutti tranne che per pochi dei problemi più semplici dell’idrodinamica. La teoria della turbolenza, lo studio delle onde d’urto, la teoria delle

²⁸⁹ “Steel tape and other similar devices for this purpose are almost independent of the particular nature of the problem to be solved”.

²⁹⁰ “Wiener’s realization of Turing’s “tape” as a multitrack magnetic tape in which the “bits” of a binary number are recorded simultaneously on parallel tracks came at least a quarter century ahead of the industry”.

²⁹¹ To Wiener, the electronic computer was a prosthetic aid to enable the human mind to transcend its limitations and better perform its duties.

esplosioni, la balistica interna, lo studio del moto di un proiettile a velocità supersonica, ecc.”. Ambiti determinati da una complessità tale da non avere nessun “controllo computazionale” e nessuna possibilità di sapere “se la teoria concordi con la pratica” [Wiener, 1940 trad. mia]²⁹²

Il legame delineato tra la capacità del dispositivo di eseguire compiti oltre a portata computazionale della mente con la dimensione materiale, che si accorda con il residuo semantico persistente nella teoria dell’informazione come riduzione dell’entropia elaborata dalla cibernetica wieneriana [Kline, 2015, p. 15], può contribuire a chiarire il giudizio espresso da Stanislaw Ulam sulle diverse “filosofie” della computazione incarnate da Wiener e von Neumann.

Se quest’ultimo, ma vedremo come il pensiero di von Neumann subisca un’evoluzione su questo punto, si attestava su una concezione ortodossa della computazione come sistema “digitale o binario o puramente discreto”, Wiener, al contrario, riteneva che i computer del futuro sarebbero stati più simili a macchine analogiche che a dispositivi digitali. Tale convinzione, secondo Ulam, derivava dalla centralità che Wiener attribuiva alla natura organica - e in particolare alla funzione ormonale - del cervello [Ulam, 1980, p. 94, trad. mia]²⁹³. Idee che si ritroveranno poi in *Cybernetics*, in cui viene ipotizzato il ruolo degli ormoni nella comunicazione nel sistema neurofisiologico visto come un meccanismo a *feedback* negativo [Wiener, 1948, pp. 173-175 trad. it.].

Bush tuttavia scrisse a Wiener il 19 ottobre del 1940 , malgrado la stima personale e l’apertura alle sue idee innovative²⁹⁴, riferendogli che malgrado “argomento della macchina per

²⁹² “If machines of this sort can be devised, they will be of particular use in many domains in which the present theory is computationally so complex as to be nearly useless. This is true of all but a few of the simplest problems in hydrodynamics. Turbulence theory, the study of waves of shock, the theory of explosions, internal ballistics, the study of the motion of a projectile above the speed of sound, etc., suffer greatly for the lack of computational tools. There are many cases where our computational control is so incomplete that we have no way of telling whether our theory agrees with our practice”.

²⁹³ “A little later these two men had apparently developed different philosophies. Wiener thought that the computers (if you could use the word at that time) would be more in the nature of analog machines than digital. Von Neumann maintained the opposite view. Wiener thought of the hormonal activity of the human brain- obviously there is no mechanical relay system in our brain-so he thought that the big developments would go in the direction of some kind of system of fluids, whereas von Neumann from the start was thinking of developing a digital or binary or purely discrete system”.

²⁹⁴ Il memorandum di Wiener, se fosse stato diffuso nei centri come Harvard o Philadelphia, dove si stavano sviluppando forme di computazione digitale, poteva modificare la storia dello sviluppo del calcolatore. Figure di primo piano della creazione del computer, come Goldstine, ne hanno avuto visione solo nel 1982 [Masani, 1990, p.175]. La parte finale della nota di Wiener ipotizza alcune caratteristiche per un uso generale della macchina delineata dal memorandum: “Per una macchina di applicabilità generale, dobbiamo disporre di: 1 un apparato a nastro simile a quello della macchina potenziale, ma eventualmente con più fili per gestire un numero maggiore di funzioni interne e dati al contorno; 2 un apparato di scansione e registrazione per effettuare letture e registrazioni in più punti; 3 macchine elettroniche capaci di eseguire rapidamente sequenze di operazioni, come somma e moltiplicazione, sui dati letti, prima di stampare il risultato in scala binaria. [“For a machine of general applicability, we must have: 1 a tape apparatus like that of the potential machine, only perhaps with more wires for more

le equazioni differenziali parziali non [fosse] stato dimenticato” egli, come capo dell’NDRC, non era “ancora stato convinto che sia giustificabile come programma per il *National Defense Research Committee* in relazione alla difesa nazionale” [Bush, lettera a N. Wiener, 19 ottobre, 1940 trad. mia]²⁹⁵.

Wiener dovette trovare un’altra “specie di angolo dove essere utile durante l’emergenza” [Wiener, lettera a V. Bush, 21 settembre 1940 trad. mia]²⁹⁶. In questo altro angolino, nacque la cibernetica.

Vale la pena di ricordare un’altra, “incendiaria”, proposta di Wiener di contributo bellico, precedente al predittore, contenuta nella lettera a Bush. L’idea di Wiener era quella di creare “una barriera contraerea facendo esplodere in aria contenitori di etilene, propano o acetilene liquefatti, in modo che una regione significativa dell’aria venga riempita con una miscela esplosiva di questi gas” [Wiener, lettera a V. Bush, 21 settembre 1940 trad. mia]²⁹⁷. Bush su questo punto non rispose mai a Wiener.

Gli rispose, invece, il 31 dicembre del 1940, con una lettera in cui respingeva la proposta di costruzione di una macchina calcolatrice per le PDE. Bush, pur apprezzando il progetto aveva “la netta sensazione che il tipo di competenze necessarie per perfezionare un dispositivo del genere nei suoi aspetti meccanici sia attualmente fortemente impegnato in vari progetti di ricerca legati alla difesa”. Pur presentando motivi di interesse bellico, il dispositivo di Wiener “appartiene chiaramente a una categoria di progetti a lungo termine, e sembra essenziale, al momento, che le persone particolarmente qualificate in questo settore vengano impiegate il più possibile su questioni di più immediata rilevanza” [Bush, lettera a N. Wiener, 31 dicembre 1940,

internal functions and boundary data; 2 scanning and recording apparatus for scanning and recording in more places; and 3 electronic machines capable of performing rapid sequences of operations such as addition and multiplication on the data read off, before printing the result on the binary scale. The machines in (3), while perhaps expensive, are not often repeated, and their total cost need not be excessive”. Wiener, 1940]. Ferry e Seaks, nel loro commento al Memorandum, arrivano a dire che se questo fosse circolato ora parleremmo di macchine di Wiener al posto di quelle di von Neumann: “Yet, had Bush circulated the Wiener memorandum, we might today be talking about the limitations of the Wiener–von Neumann, if not the Wiener machine, instead of the von Neumann machine” [Ferry, Seaks, 1987, p.196].

²⁹⁵ “I write merely to tell you that the subject of the partial differential equation machine has not been forgotten. I have not yet been persuaded that it is justifiable as a program for the National Defense Research Committee in connection with the national defense”.

²⁹⁶ “I enjoyed very much the chance to talk with you on your visit here and hope you can find some corner of activity in which I may be of use during the emergency”.

²⁹⁷ “There is another quite different matter which I mentioned to you in private conversation and which I would like to repeat for purposes of record. It has to do with the idea of making an anti-aircraft barrage by bursting in the air containers of liquified ethylene or propane or acetylene gases so that an appreciable region will be filled with an explosive mixture of those in the air. The idea is that in such a way a region of the air may be interdicted to enemy aircraft—a matter of tens of seconds or of minutes instead of the fraction of a second during which a high explosive burst is continuing, and that thereby the efficiency of a barrage from a small number of anti-aircraft guns may be appreciably increased”.

trad. mia]²⁹⁸. Bush non seppe cogliere che sarebbe stata proprio la guerra a costituire il principale fattore trainante – *esterno* [Shapin, 1992] - nello sviluppo di un calcolatore che supererà il paradigma rappresentato dal suo *Differential Analyzer*.

3. Più forte delle bombe: il superamento dell'analogico e l'informazione digitale

La macchina di Bush, come è stato affermato sopra, era inefficace nei confronti delle PDE. Era progettata per calcolare quelle ordinarie. Rappresentava il modo “pratico” delle idee già elaborate da Lord Kelvin intorno risoluzione delle equazioni differenziali [Goldstine, 1972, p,51]. Le equazioni che l'Analyzer era in grado di risolvere riguardavano, ad esempio, il moto di un corpo proiettato verticalmente verso l'alto in un campo gravitazionale costante, assumendo che la resistenza dell'aria fosse proporzionale alla velocità:

$$\frac{d^2x}{dt^2} + k \left(\frac{dx}{dt} \right) + g(x) = 0$$

Questa equazione può essere inserita nella macchina una volta presa la forma, che ne rende più evidente la connessione, di:

$$\frac{dx}{dt} = - \int \left(k \frac{dx}{dt} + g \right) dt$$

La procedura per inserire l'equazione nella macchina è, grosso modo, la seguente: a ciascuna grandezza significativa presente nell'equazione viene assegnato un albero di trasmissione (*bus shaft*). Le varie relazioni esistenti tra queste grandezze vengono poi stabilite mediante connessioni alle unità operative: una relazione funzionale si ottiene collegando i due alberi corrispondenti a un piatto di ingresso (*input table*), una somma si rappresenta posizionando un sommatore (*adder*), una relazione di integrazione tramite un integratore, e così via. Quando tutte le relazioni coinvolte sono state

²⁹⁸ “I believe that such a machine would be valuable, and I am inclined to believe that it could be successfully constructed. In examining into the matter, however, I feel sure that the type of talent needed to perfect such a device in its mechanical aspects is bound to be heavily engaged at the present time on defense research matters of various sorts. While your device would also be of aid on defense matters, it is undoubtedly of the long-range type and it appears essential that at the present time those individuals who are particularly qualified along these general lines be employed as far as possible on matters of more immediate promise. Accordingly, I do not feel justified in recommending active work on the subject right now. I hope that later on we can again pick up the subject”.

rappresentate in questo modo, si effettua una connessione finale che rappresenta l'uguaglianza espressa nell'equazione. [Bush, 1931 trad. mia]²⁹⁹

L'Analizzatore "non manipolava numeri, lavorava su grandezze" [Gleick, 2010, p.162 trad. it.]. Era una macchina *analogica*. "In un calcolatore analogico tutti i numeri sono rappresentati da grandezze fisiche opportune, i cui valori, in termini di unità di misura prefissate, sono uguali ai numeri in questione. Si usano, ad esempio, l'angolo di rotazione di un dato disco; l'intensità di una data corrente, l'entità di una data tensione relativa ecc." [von Neumann, 1958, p.108 trad. it. 1994].

Un regolo calcolatore oppure "l'ombra gettata dallo gnomone sulla superficie di una meridiana, il cadere tendenzialmente regolare della sabbia in una clessidra o dell'acqua in un orologio ad acqua, o l'oscillazione matematicamente costante di un pendolo" [Floridi, 2010, p. trad. it] sono buoni esempi elementari di calcolatori analogici. I numeri in essi sono rappresentati come grandezze fisiche: "Il progettista di un dispositivo analogico decide quali operazioni intende eseguire, e poi cerca un apparato fisico le cui leggi di funzionamento siano analoghe a quelle che vuole realizzare. Successivamente costruisce l'apparato e risolve il problema misurando le grandezze fisiche - e quindi continue - coinvolte nell'apparato stesso" [Goldstine, 1972, p.70, trad. mia]³⁰⁰.

Intimamente legato al concetto di "*analogo*" è quello di *continuo*. Un calcolatore analogico elabora - o rappresenta - dati che possono assumere qualsiasi valore in un intervallo continuo, ovvero valori appartenenti all'insieme dei numeri reali. [Cartuyvels et al., 2021]. Il calcolo analogico si basa su una corrispondenza diretta tra grandezze fisiche (come le tensioni, le rotazioni o gli spostamenti) e variabili reali, permettendo una rappresentazione continua dell'informazione.

L'Analizzatore Differenziale di Bush [fig.14] risolveva equazioni differenziali traducendo variabili e derivate in rotazioni di alberi meccanici: l'integrazione avveniva fisicamente tramite ingranaggi, dischi e ruote che, opportunamente collegati, producevano movimenti fisicamente

²⁹⁹ "The procedure of placing the equation on the machine is somewhat as follows: A bus shaft is assigned to each significant quantity appearing in the equation. The several relations existing between these are then set up by means of connections to the operating units: a functional relation by connecting the two corresponding shafts to an input table, a sum by placing an adder in position, an integral relationship by an integrator, and so on. When all the relationships which are involved have been thus represented a final connection is made which represents the equality expressed in the equation".

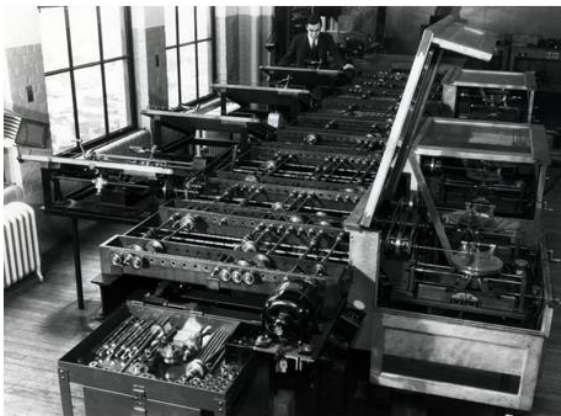
³⁰⁰ "The designer of an analog device decides what operations he wishes to perform and then seeks a physical apparatus whose laws of operation are analogous to those he wishes to carry out. He next builds the apparatus and solves his problem by measuring the physical, and hence continuous, quantities involved in the apparatus".

analoghi ai valori calcolati [Bush, 1931; Goldstine, 1972, pp.84-99]. Le ruote, gli ingranaggi, i dischi, i carrelli, gli alberi meccanici che lo componevano servivano a produrre “un analogo fisico delle equazioni differenziali” [Gleick, 2010, p.162 trad. it.].

La risoluzione delle equazioni differenziali della macchina di Bush aveva una accuratezza del 2% [Hartree, 1940]. Una copia della macchina di Bush era stata acquistata nel 1935 dal *Ballistic Research Laboratory* con sede all'*Aberdeen Proving Ground* in Maryland. Proprio le richieste computazionali di questo luogo hanno rappresentato il maggior vettore di sviluppo per i calcolatori negli Stati Uniti [Goldstine, 1972, p.135].

Oswald Veblen (1880–1960) – capo del dipartimento di matematica di Princeton e tra i fondatori dell'*Institute for Advanced Study* – durante la Prima guerra mondiale assunse il grado di maggiore e, nel gennaio 1918, fu nominato responsabile dell'*Aberdeen Proving Ground* [Barber, 1956, p.29; Dyson, 2012, pp.21-26 trad.it.]. Veblen comprese che, per affrontare i complessi problemi legati ai calcoli balistici e aerodinamici, era necessario rivolgersi a figure con una solida preparazione nell'analisi matematica, in particolare nella risoluzione delle equazioni differenziali. Queste figure erano una “razza debole”, considerati in patria “come una massa di goffi e inutili trafficanti di simboli” [Wiener, 1953b, p.255]³⁰¹. Erano i matematici dell'accademia americana.

Fig.14 Il Differential Analyzer di Bush, e il suo creatore



³⁰¹ “Thus we American mathematicians were a feeble race, and the country at large regarded us as so many useless fumblers with symbols”.

I matematici più prestigiosi degli Stati Uniti confluirono all'*Aberdeen Proving Ground* sotto la direzione di Oswald Veblen. "Durante la Prima guerra mondiale, per la scienza americana Aberdeen fu quello che Los Alamos sarebbe stato una generazione dopo: un luogo in cui le menti matematiche più esperte, vecchie e nuove, sposarono una missione e un cameratismo a quel tempo sconosciuti" [Conway & Siegelman, 2005, p.62 trad. it.].

Furono proprio questi matematici ad essere chiamati "computer", come lo sarebbero stati più di vent'anni dopo da John von Neumann a Los Alamos, dove il termine indicava lo staff - una ventina di persone - incaricato di eseguire i calcoli necessari agli esperimenti nucleari. Allo stesso modo, a *Bletchley Park*, lavorava il matematico, Alan Turing, che aveva formulato la nozione rigorosa di *algoritmo* proprio a partire dalla rappresentazione estremamente schematizzata di un essere umano impegnato in calcoli con carta e matita.

Tra questi computer umani, che sull'onda dell'entusiasmo per il lavoro compiuto aveva provato, con risultati tra il comico e il tragico, ad arruolarsi nell'esercito, c'era anche Norbert Wiener³⁰². Tuttavia le anche le prestazioni dei computer umani si rivelarono ben presto insufficienti di fronte ai progressi della scienza balistica del Novecento. C'era bisogno che il computer di carne fosse affiancato da uno di metallo per produrre ciò per cui il Ballistic Research Laboratory era nato: le tabelle di tiro [Goldstine, 1972, p.135]³⁰³.

Sul campo di battaglia la sola conoscenza delle traiettorie ha importanza quasi nulla per l'artigliere.

Ciò di cui aveva bisogno era una tabella di tiro che indicasse la gittata attesa per ciascun cannone e obice, quando sparava un determinato proiettile con una specifica carica propellente a un dato angolo di elevazione. La tabella doveva anche indicare le correzioni da applicare per le variazioni di

³⁰² Le disavventure del soldato Wiener sono narrate da egli stesso nella sua autobiografia [Wiener, 1953a, pp. 254–263]. Dopo il lavoro come matematico al Proving Ground, e animato da un fervente patriottismo, Wiener fu ammesso nell'esercito nell'autunno del 1918. Malgrado avesse riconosciuto in una lettera al padre Leo il proprio "talento speciale nel combinare pasticci" [Conway & Siegelman, 2005, p.63 trad. it.], affrontò con entusiasmo l'arruolamento. Due giorni prima della firma dell'armistizio fu inviato, ironia della sorte, proprio al campo di addestramento di Aberdeen. Tuttavia, come lui stesso racconta, ebbe presto di che pentirsi, e l'esperienza accentuò uno stato depressivo che lo avrebbe accompagnato a lungo: "Ero inorridito dall'irrevocabilità del passo che avevo compiuto. Mi sentivo come se fossi stato condannato al penitenziario" ["I felt as if I had been sentenced to the penitentiary", Wiener, 1953a, p. 259]. La sua uniforme, ricorda, "era tesa per la mia corpulenza" ["My uniform was strained by my corpulence", Wiener, 1953a, p.259], e il suo aspetto contribuì a renderlo un bersaglio facile per gli altri commilitoni. Divenne presto lo zimbello del campo, vittima di frequenti scherzi: in un'occasione fu persino convinto a tagliarsi i suoi amati baffi [Conway & Siegelman, 2005, p.64].

³⁰³ "The main functions of the Ballistic Research Laboratory was the production of firing and bombing tables and related gun control data".

temperatura atmosferica, densità dell'aria, vento, angolo di sito, peso del proiettile, velocità alla volata e compensazioni per la deriva e, in alcuni casi, per la rotazione della Terra. [Barber, 1956, p.3]³⁰⁴.

Solo tenuto conto nel calcolo delle tabelle di tiro di tutti questi fattori si potevano elaborare le tabelle di tiro. Queste tabelle avevano a volte la forma "di piccoli libretti che entravano nella tasca dell'artigliere oppure come automi - piccoli calcolatori analogici altamente specializzati - collegati ai cannoni, i quali accettavano come input la posizione radar e le informazioni secondarie, e posizionavano i cannoni senza intervento umano". Questi ultimi, i *director*, furono introdotti durante la Seconda Guerra Mondiale a causa della maggiore velocità dei velivoli nemici da colpire. Tuttavia questo automa si limitava a incorporare le tabelle che andavano comunque calcolate. "Le tabelle di bombardamento e i mirini erano del tutto analoghi" [Goldstine, 1972, p. 136]³⁰⁵.

Un calcolatore umano, dotato di calcolatrice da tavolo, impiegava più o meno 10 secondi per eseguire una moltiplicazione. Il calcolo di una traiettoria richiedeva all'incirca 750 moltiplicazioni. Ciò significa che per compiere i calcoli di una singola traiettoria servivano 7.500 secondi, equivalenti a due ore. A questo tempo andavano aggiunti i tempi relativi alla scrittura su carta dei risultati della moltiplicazione, alla gestione dei dati: bisognava moltiplicare allora il tempo per un fattore 6 [Goldstine, 1972, pp. 137-138; Dyson, 2012, trad. it., pp.81-82].

Per calcolare una traiettoria servivano allora 12 ore. Una tabella di tiro era composta in media da 750 traiettorie da calcolare. Un calcolatore umano, nell'ipotesi lavorasse senza interruzioni 24 ore al giorno e sette giorni su sette, impiegherebbe 9.000 ore, ovvero 375 giorni: più di un anno. Ovviamente questi erano tempi insostenibili con le esigenze balistiche. Automatizzare e velocizzare i processi era diventato un imperativo.

³⁰⁴ "Information about projectile trajectories, however accurately computed, had little direct value for the artilleryman in the field. What he needed was a firing table that showed the range to be expected for each gun and howitzer when it fired a specific projectile with a given propelling charge at any selected angle of elevation. The table also had to indicate the corrections to be applied for variations in atmospheric temperature, air density, wind, angle of site, weight of projectile, muzzle velocity, and compensation for drift, and in some cases, for the rotation of the earth".

³⁰⁵ "These tables sometimes existed as small booklets that fitted into a gunners pocket or as automata-small highly specialized and usually analog computers-attached to guns, which accepted as input radar locations plus the secondary information and without human intervention positioned the guns. These devices were introduced during World War II because of the speed requirements imposed as a result of airplane targets. However, the same material needed to be calculated in any case since the director, as the fire control apparatus was called, had built into it the ballistic tables and merely looked up entries with great speed and used this information to control electrical motors. Bombing tables and bombsights were entirely analogous".

All'inizio degli anni Trenta, il dottor Vannevar Bush, del Massachusetts Institute of Technology, sviluppò un calcolatore per risolvere equazioni differenziali tramite integrazione meccanica. Nel 1934, il dottor Bush propose che questo calcolatore venisse utilizzato per calcolare traiettorie balistiche, ma a quel tempo la mancanza di fondi impedì di avviare il lavoro. Un anno dopo, anche il capitano Alger suggerì l'uso dell'analizzatore per questo scopo. La sua raccomandazione fu approvata dal colonnello Shinkle, comandante dell'Aberdeen Proving Ground, e vennero presi accordi affinché il dottor L. S. Dederick, R. H. Kent e il professor A. A. Bennett visitassero il Massachusetts Institute of Technology per assistere a una dimostrazione. Poco dopo, furono presi accordi per la costruzione di un Analizzatore Differenziale di Bush destinato all'Aberdeen Proving Ground. La macchina fu pronta per l'uso nel 1935. Fu impiegata nella preparazione delle tavole di tiro e il lavoro necessario per produrle fu enormemente ridotto. I due grandi vantaggi dell'analizzatore erano la velocità e l'accuratezza, e il suo successo segnò l'inizio dello sviluppo di calcolatori specializzati per i vari tipi di calcoli balistici [Barber, 1946, p.16 trad. it. mia]³⁰⁶.

L'utilizzo dell'Analizzatore Differenziale aveva in parte assolto al compito di rendere più rapida e precisa la produzione delle tavole di tiro. "La necessità di tabelle balistiche esterne era stata notevolmente ridotta grazie all'acquisizione dell'Analizzatore Differenziale di Bush da parte di Aberdeen [...] l'analizzatore di Bush rese possibile la compilazione diretta delle tabelle di tiro, in modo molto più rapido rispetto a qualsiasi metodo precedente" [Barber, 1956, p. 21, trad. it. mia]³⁰⁷.

Tuttavia ben presto le prestazioni dell'Analizzatore - che impiegava dai 10 ai 20 minuti per calcolare una traiettoria con una precisione di circa 5 parti su 10.000 [Goldstine, 1972, p.], sebbene quando installato rappresentasse lo strumento più importante mai acquistato dal centro di Aberdeen, non riusciva a tenere il passo con la mole di lavoro che gli veniva richiesta [Barber, 1956, p.21 trad. it. mia].

³⁰⁶ "In the early 1930s Dr. Vannevar Bush, of the Massachusetts Institute of Technology, developed a computer for solving differential equations by mechanical integration. In 1934, Dr. Bush suggested that this computer be used to calculate trajectories, but at that time lack of funds prevented the work. A year later Captain Alger also suggested the use of the analyzer for this purpose. His recommendation was approved by Colonel Shinkle, Commanding Officer of Aberdeen Proving Ground, and arrangements were made for Dr. L. S. Dederick, R. H. Kent, and Professor A. A. Bennett to visit Massachusetts Institute of Technology to witness a demonstration. Shortly thereafter arrangements were made for construction of a Bush Differential Analyzer for Aberdeen Proving Ground. The machine was ready for use in 1935. It was used in the preparation of firing tables, and the work required for the tables was reduced enormously. The analyzer's two great advantages were speed and accuracy, and its success marked the beginning of the development of specialized computers for ballistic computations of various kinds".

³⁰⁷ "The need for exterior ballistic tables had been greatly reduced by Aberdeen's acquisition of the Bush differential analyzer. As originally planned, the ballistic tables were to be used for compiling firing tables for specific gun-ammunition combinations; however, the Bush analyzer made possible the compilation of firing tables directly and much more rapidly than could be done by any older method".

Vi erano delle difficoltà strutturali proprie di ogni macchina analogica. Un calcolatore analogico ha la sua essenza in un'idea: quella per cui, a diversi gradi di complessità, "data una formula matematica, potevano ideare una macchina esattamente descrivibile da quella formula". Questo tipo di macchine presenta tre limiti: "la generalità, la precisione e la velocità" [Goldstine, p.140, trad. mia]³⁰⁸.

Nel caso in cui una macchina equivalga a una descrizione matematica, si incontrano inevitabilmente problemi di natura ingegneristica: "ogni macchina è costruita con componenti che hanno determinate tolleranze e possono funzionare solo entro certe velocità". Questo fattore va a detrimento tanto dell'accuratezza quanto della velocità. La dimensione fisica delle parti della macchina non solo può operare solo entro certi ordini di velocità - imposti dalla meccanica newtoniana - ma è anche soggetta a degrado: "col passare del tempo, man mano che le parti perdono precisione, la risposta complessiva della macchina diventa meno affidabile. Inoltre, è spesso vero che, più velocemente si fa funzionare un dispositivo meccanico, meno accuratamente esso rappresenta la situazione matematica" [Goldstine, 1972, pp.140-141 trad. mia]³⁰⁹. Precisione e rapidità entrano così in conflitto. La macchina di Bush era tanto più precisa quanto più lentamente veniva fatta funzionare.

La velocità, inoltre, era compromessa da un'ulteriore questione: l'Analizzatore - in tutte le sue evoluzioni - doveva essere riprogrammato per ogni singolo compito, con un ulteriore spreco di tempo. La macchina analogica, allora, non poteva essere generale - in termini turinghiani, universale - nel senso che sarà esaminato più avanti. Funzionava - e poteva funzionare - soltanto per compiti computazionali specifici. "Data la loro natura fisica, i computer fisici operano in 'tempo reale' (nel senso del concetto di tempo nel mondo reale) e per osservare e controllare l'accadere di eventi in una relazione 1 a 1 tra il tempo dell'evento e il tempo del calcolo (si pensi alla clessidra)". Non sono macchine generali, "funzionano esclusivamente come dispositivi necessariamente specializzati" [Floridi, 2010].

³⁰⁸ "We noted earlier that the nineteenth-century physicists had attained an ability to describe in mathematical form quite complex machinery and were therefore able to invert this process. Given a mathematical formula, they were, in principle at least, able to invent a machine exactly describable by the formula. This is what the analog computer is. What are the limitations on this idea? There are fundamentally three aspects to the problem having to do with the generality, the accuracy, and the speed of analog equipment".

³⁰⁹ "Given a machine whose operation is describable by a mathematical formula, we quickly run up against two realities of engineering: every machine is built out of parts having certain tolerances and capable of running at certain speeds. These determine the accuracy and speed with which the mathematical system can be solved. Not only is there an upper limit on how exactly one can smooth physical surfaces, cut gears, etc., there is also a degradation that sets in with age when the parts become less accurate, so that the precision of the total machine's response diminishes. Moreover, it is often true that the faster a mechanical device is driven the less accurately it depicts the mathematical situation, with the result that often accuracy and speed are linked together in a disastrous, or at least unhappy, way"

I calcolatori analogici - le macchine che *misurano* [Wiener, 1950, p. 91 trad. it.; Goldstine, 1972, p. 39; Bruderer, 2020, p. 39] – erano limitati nella loro stessa struttura computazionale. Le esigenze di calcolo dovevano allora orientarsi verso un altro paradigma: quello delle macchine *numeriche*, che *contano*. In queste macchine, le operazioni di computo fondamentali - addizione, sottrazione, moltiplicazione e divisione - vengono meccanizzate, e si lavora col familiare metodo di rappresentare i numeri come aggregati di cifre [*digits*] [von Neumann, 1951, p. 7; Goldstine, 1972, p.136, p. 39]. Queste macchine non erano soggette alle limitazioni dei calcolatori analogici. Per quale motivo?

Per il motivo per cui le operazioni matematiche possono essere ridotte a operazioni elementari, cioè a quelle “che la macchina considera elementari, ovvero che il calcolatore può eseguire direttamente; tutte le definizioni implicite devono essere riformulate in modo esplicito mediante procedure finite e costruttive; e i processi al limite devono essere troncati e sostituiti con sequenze finite di operazioni elementari”. Un calcolatore considera elementari solo le operazioni che sono integrate, incorporate [*built-in*] nella sua struttura. Una macchina analogica come quella di Bush aveva in sé come operazioni elementari l'integrazione, la risoluzione e il differenziare necessario alla risoluzione delle equazioni differenziali [Goldstine, 1972, p.142]³¹⁰. Il paradigma numerico, invece, ha come proprie operazioni elementari incorporate quelle imitano i processi di un essere umano impegnato in operazioni di calcolo

Questo approccio, non rappresentando le quantità mediante grandezze fisiche analoghe, si libera dai vincoli imposti dalla fisica - e dalle limitazioni dell'ingegneria - alla misurazione: la rappresentazione discreta, assumendo valori numerici precisi, riduce la sensibilità alle imprecisioni fisiche e garantisce una maggiore affidabilità computazionale, poiché lavora con

³¹⁰ “However, once it is decided to handle the problem by numerical calculation on a particular kind of machine then all transcendental operations must be replaced by elementary ones -those operations the machine views as elementary, i.e. which the computer can handle directly; all implicit definitions must be replaced by explicit ones involving finite and constructive procedures; and limiting processes must be truncated and replaced by finite sequences of elementary operations. The operations that a given computer views as elementary are those that are “built in.” Thus, the Bush differential analyzer has as its elementary operations differentiating, integrating, and solving certain (essentially implicit) differential equations. Un’efficace sintesi delle operazioni elementari di una macchina come quella di Bush si trova in von Neumann [1958, p.109 trad. it.1994]: “Così, l’analizzatore differenziale classico, in cui i numeri sono rappresentati mediante angoli di rotazione di dischi, funziona nella seguente maniera. Invece dell’addizione, $(x + y)$, e della sottrazione, $(x - y)$, esso effettua le operazioni $(x \pm y)/2$, per mezzo di un organo semplice e di agevole impiego, l’ingranaggio differenziale (lo stesso che viene utilizzato nelle automobili). Alla moltiplicazione, $x \cdot y$, si sostituisce un procedimento del tutto differente: nell’analizzatore differenziale tutte le grandezze si presentano come funzioni del tempo, e l’apparecchio fa uso di un organo, detto “integratore”, che può - date due grandezze $x(t)$, $y(t)$ - calcolare l’integrale di Stieltjes: $z(t) = \int x(t) dy(t)$.”

variabili che possono assumere un numero limitato o numerabile di valori distinti [Cartuyvels et al., 2021].

Questi valori sono i numeri che possono essere computati. I calcolatori di questo tipo “possono essere classificati tra le «macchine a stati discreti», cioè tra quelle che si muovono a salti o scatti improvvisi da uno stato ben definito a un altro [...]. Strettamente parlando non esistono macchine del genere: in realtà ogni cosa si muove con continuità. Ma ci sono molti tipi di macchine che possono vantaggiosamente essere viste [*thought*] come macchine a stati discreti” [Turing, 1950, p. 129 trad. it.].

Il fabbisogno di produzione automatizzata di tavole di tiro nei centri balistici statunitensi non poteva più essere soddisfatte da una macchina a paradigma analogico come quella di Bush. Le innovazioni tecnologiche nella raccolta dei dati avevano prodotto sviluppi ad Aberdeen, come la telecamera balistica [*ballistic camera*] - un sistema sincronizzato di cineprese, ideato nel 1941 e concretamente impiegato dal 1943, inizialmente usato solo di notte e poi anche di giorno grazie all'introduzione di lampade *flash* ad alta intensità - per registrare con precisione la traiettoria e il punto d'impatto di bombe e razzi tramite segnali luminosi e dispositivi cronografici. L'analizzatore analogico di Bush non era più in grado di processare la mole di dati prodotta da queste innovazioni.

Aberdeen si rivolse allora alla *Moore School of Engineering* di Philadelphia [Barber, 1956, p.38], dove si stavano progettando nuovi tipi di macchine (che analizzeremo nello specifico più avanti) maggiormente capaci di soddisfare, dal punto di vista della rapidità e dell'accuratezza, le esigenze computazionali che le necessità militari imponevano.

Questo tipo di macchine sono le macchine *digitali*. L'automatizzazione dei processi di produzione di tabelle di tiro è stata la “*raison d'être*” dello sviluppo dei computer digitali [Goldstine, 1972 p. 135]³¹¹.

Nella primavera del 1942, a guerra mondiale in corso, una commissione scientifica dell'NDRC - che, come abbiamo visto in precedenza, era l'ente a capo della ricerca scientifica in campo militare - si trovò a dover valutare una serie di proposte progettuali di strumenti balistici

³¹¹ The automation of this process was to be the *raison d'être* for the first electronic digital computer. Herman Goldstine (1913–2004), matematico, ufficiale dell'esercito e pioniere dell'informatica statunitense – figura centrale nello sviluppo della computazione – fu tra i principali responsabili del progetto ENIAC, collaborò alla progettazione dell'EDVAC e diresse lo sviluppo del computer dello IAS insieme a John von Neumann, di cui divenne stretto collaboratore. Nel suo fondamentale volume di storia del computer, *The Computer from Pascal to von Neumann* [Goldstine, 1972], fa esplicito riferimento proprio all'ENIAC come al primo calcolatore elettronico digitale, sebbene non ancora a programma memorizzato. Tuttavia, la successiva desecretazione di progetti britannici condotti durante la Seconda guerra mondiale – come vedremo – rimetterà in discussione questo primato.

antiaerei. Questi potevano essere divisi in due macro-gruppi: quelli ispirati al paradigma analogico e quelli costruiti secondo il paradigma digitale, che abbiamo appena esaminato. George Stibitz, già incontrato come protagonista dell'episodio perturbante con il predittore wieneriano, membro della commissione, sentì che il termine "impulso" [*pulse*], usato per descrivere il funzionamento delle macchine calcolatrici numeriche elettriche, non fosse adeguato. "Propose quindi un altro termine che gli sembrava più descrittivo: *digitale*" [Ceruzzi, 2012, p. 1]³¹².

Detto ciò, possiamo ora essere più precisi su cosa si intenda con approccio digitale. Esso consiste nella consapevolezza che si può costruire una macchina capace di imitare il metodo umano di calcolo: contare e costruire le operazioni elementari - somma, sottrazione, moltiplicazione, divisione - tramite il conteggio. Non solo questo è possibile, ma si può anche dimostrare che, in generale, le formulazioni matematiche possono essere trattate per mezzo di queste operazioni elementari [Goldstine, 1972, p.142 trad. mia]³¹³.

In una macchina digitale si automatizzano le operazioni aritmetiche basilari [von Neumann, 1951 p.7] di computo su "grandezze numeriche con valori definiti (a gradini) (senza valori intermedi)" [Brubeder, 2020, p.38, tab. 2.1 trad. mia]³¹⁴. Come si rappresentano queste grandezze in modo digitale?

Il termine inglese "*digit*", che significa "cifra", deriva dal latino *digitus*, "dito", e trova origine dall'atto di contare con le dita. Tre componenti fondamentali nella formazione del concetto e della rappresentazione di numero, e cioè, capacità innata di rappresentare piccole numerosità [*numerocities*]; la capacità di usare funzionalmente le proprie dita; e la capacità di rappresentare mentalmente le proprie dita, sarebbero legate proprio all'atto di contare [Butterworth, 1999; Penner-Wilger et al. 2007].

Nello sviluppo cognitivo del bambino, le dita costituirebbero possono essere addirittura considerate come l'anello mancante [*missing link*] tra le rappresentazioni preverbal dei numeri e le parole che indicano questi ultimi. [Fayol, Seron, 2005; Penner-Wilger et. al, 2007]. La

³¹² One member of the committee, Bell Telephone Laboratories mathematician George Stibitz, felt that the term *pulse* was not quite right. He suggested another term that he felt was more descriptive: *digital*.

³¹³ "These things having been said, we may now be more precise about what is meant by the digital approach. It is the realization that a machine can be built to imitate the human method of calculating: to count and to build up the elementary operations-addition, subtraction, multiplication, division-by counting. Not only can this be done but it may be shown that, in general, mathematical formulations may be handled by means of these elementary operations."

³¹⁴ "Numerical magnitudes with definite(steped) values (without intermediate values)"

diffusione, sino a diventare ordinaria, della rappresentazione numerica in base 10 dipenderebbe allora dal numero di dita delle nostre mani: “la preponderanza della base 10 è dovuta al fatto contingente che abbiamo 10 dita” [Deheane, 2011, p.117]³¹⁵.

Anche Wiener, in *Cybernetics*, nota come, nel diffondersi di calcolatrici a base 10, “il nostro impiego di macchine a sistema decimale è dovuto semplicemente alla contingenza storica per cui questo sistema, basato sul numero delle nostre dita, era già in uso quando gli indiani fecero la grande scoperta dell'importanza dello zero e dei vantaggi di un sistema di notazione posizionale” [Wiener, 1948, p.61 trad. it.]

In linea di principio, malgrado la familiarità con la notazione decimale, quest'ultima non è necessaria a una macchina digitale [von Neumann, 1951, p.7]. Assegnando a ciascuna cifra decimale, da 0 a 9, un segnale [*marker*], è possibile rappresentarle mediante un sistema in cui ogni marker assuma una tra dieci forme differenti.

Tuttavia, nei sistemi automatizzati di computazione che operano mediante segnali binari, rappresentare le 10 cifre richiede almeno 4 segnali binari: infatti, 3 segnali binari producono solo $2^3 = 8$ combinazioni, che sono insufficienti. 4 segnali, invece, producono $2^4 = 16$ combinazioni, che superano il minimo necessario. “Un gruppo di tre segnali a due possibilità permette otto combinazioni, ed è quindi insufficiente; un gruppo di quattro permette sedici combinazioni, il che è più di quanto occorra strettamente. Quindi per ogni cifra decimale vanno usati gruppi di almeno quattro segnali” [von Neumann, 1958, p.110 trad. it.1994].

Entrambe le alternative di rappresentazione in base 10, sia il marker a 10 possibilità e sia quello binario per 4 combinazioni, “sono altamente ridondanti” [von Neumann, 1958, p.110 trad. it.2024]. Impiegano, cioè, più combinazioni di quante strettamente necessarie per codificare 10 simboli distinti.

Nello sviluppo del paradigma digitale si è imposta, allora, la notazione in base 2: la rappresentazione *binaria*. La riflessione su questa notazione, nella scienza europea, può essere fatta risalire agli studi di Leibniz. Nella sua dissertazione *Explication de l'arithmétique binaire, où il se sert des seuls caractères 0 et 1, avec des remarques sur son utilité, et sur qu'elle donne le sens des anciennes figures chinoises de Fohy* del 1703, il filosofo tedesco, pur senza

³¹⁵ “The preponderance of base 10 is due to the contingent fact that we have ten fingers”. Nota Deheane come se le nostre mani avessero 12 dita, la base di enumerazione naturale sarebbe stata 12. E col numero delle dita, di mani e piedi, si spiegano alcune antiche notazioni in base 20 [Deheane, 2011, p.95]. Alcune interessantissime riflessioni sulle origini della notazione a base 10 ed etimologiche ed antropologiche sulle notazioni numeriche in base alle parti del corpo in diverse culture si trovano nello stesso testo del neuroscienziato francese, Deheane, 2011 e in Ifrah, 1998, pp.43-46.

raccomandare la sostituzione della notazione decimale - ormai radicata nelle abitudini di calcolo - osserva come il sistema binario, sebbene dia luogo a scritture numeriche più estese, costituisca “il modo più fondamentale di contare nella scienza”. Nell’opera, illustra inoltre come rappresentare i numeri da 0 a 32 attraverso la numerazione in base 2.

Leibniz riferisce che fu il gesuita francese residente a Pechino, Joachim Bouvet, a fargli notare la corrispondenza tra il suo sistema di calcolo binario e le figure attribuite al mitologico imperatore cinese Fu Hsi. Questi sarebbe l'autore dei trigrammi, costituiti da combinazioni di linee intere e spezzate - tradizionalmente interpretate come simboli dei principi cosmici di Yin e Yang - che, in coppia, formano i 64 esagrammi dell’*I Ching*, il *Libro dei Mutamenti* -uno dei pochissimi testi sopravvisuti a rogo dei libri ordinato dal primo imperatore Qin Shi Huang nel 213 a.C. - risalente almeno al II millennio a.C. [Leibniz, 1703; Chalmers, 2022].

Facendo un balzo in avanti e ritornando all’implementazione nelle macchine, il sistema binario comporta una serie di vantaggi: L’*hardware* del calcolatore - sia esso elettromeccanico o elettronico, ovvero l’insieme dei circuiti che processano l’informazione - implementa due soli stati fisici distinguibili: presenza/assenza di segnale, aperto/chiuso, 1/0. In tal modo, esso costituisce la traduzione materiale della notazione binaria, non solo come semplice scelta convenzionale, ma come espressione diretta delle condizioni fisiche che rendono possibile il calcolo.

Inoltre la macchina, operando in modo binario, su due soli valori che variano in modo *discreto*, è di gran lunga più precisa. Questa variazione “comporta che un computer difficilmente possa confondersi al riguardo di ciò che deve essere processato, a differenza di una macchina analogica che può non di rado funzionare in modo insoddisfacente o impreciso” [Floridi, 2010]. L’alternativa discreta tra due soli stati, “la netta differenza della decisione fra il «sì» e il «no» crea la possibilità di una accumulazione di queste decisioni, in modo tale da permetterei di discriminare piccolissime differenze in grandissimi numeri” [Wiener, 1950, p.90 trad. it.]³¹⁶.

³¹⁶ A differenza dell’analogico, limitato dai vincoli fisici del macchinario di calcolo, il metodo digitale permette di aumentare la precisione (ad esempio da 10 a 20 cifre decimali) semplicemente investendo più tempo o risorse [Goldstine, 1972, p. 143]. La velocità e il dispendio di lavoro di computazione dipenderebbero dal grado di avanzamento tecnologico dell’hardware. La macchina digitale, che ha incorporate le quattro operazioni aritmetiche, tuttavia non è sempre assolutamente precisa come può a tutta prima sembrare. L’errore può insinuarsi nelle macchine digitali a causa di eventuali limitazioni al numero di cifre che possono rappresentare, con quelle in eccesso che vengono eliminate o arrotondate. Ma l’arrotondamento pone la macchina digitale in una condizione qualitativamente simile a quella analogica: il risultato è l’output dell’operazione più un margine di errore da considerare come variabile casuale [von Neumann, 1951, p. 7]. Nei calcolatori contemporanei questa limitazione, sebbene oggi sia mitigata dallo sviluppo tecnologico, rimane a livello teorico, in quanto la rappresentazione numerica (binaria) è comunque finita: un computer può codificare solo un numero limitato di bit. Di conseguenza, troncamenti e arrotondamenti possono continuare a produrre piccoli errori nei calcoli. Questa situazione è formalmente regolata dallo standard IEEE 754, noto anche come *IEEE Standard for Floating-Point*

Questa alternativa binaria è codificata attraverso la combinazione di due soli simboli: la *binary digit*, il *bit*. Stringhe di 0 e 1. “Un *bit* è l’unità minima di informazione, niente più che la presenza o l’assenza di un segnale, 0 o 1” [Floridi, 2010]. Una stringa di *bit* forma un *byte*. Il termine è stato coniato a partire da *bite*, morso, ma riscritto per evitare che venisse accidentalmente trasformato in *bit*. [Buchholz, 1962, p.40]³¹⁷.

Tra le varie alternative inizialmente considerate (come codifiche a 6 o 12 bit), fu scelto il byte da 8 bit perché permetteva una codifica diretta in rapporto 1:1 per 256 caratteri, un intervallo ritenuto sufficiente per la maggior parte degli usi. Questa scelta semplificava la gestione dei record, ottimizzava l’uso della memoria e consentiva l’impacchettamento efficiente delle cifre decimali (due numeri a 4 bit ciascuno in un solo byte). Inoltre, poiché 8 è una potenza di 2, il byte da 8 bit agevolava l’indirizzamento e l’indicizzazione binaria fino al livello del singolo bit [Buchholz, 1962, pp. 63-66]. Una sequenza di otto bit è sufficiente per codificare le lettere dell’alfabeto latino, sia maiuscole che minuscole, le cifre numeriche, la punteggiatura e numerosi altri simboli e caratteri speciali eventualmente utilizzati [Ceruzzi, 2012, p. 170, trad. mia]³¹⁸.

La codifica standard per rappresentare lettere, numeri, simboli e segni di punteggiatura è l’*American Standard Code for Information Interchange* (ASCII), introdotta nel 1963 e adottata ufficialmente nel 1968 (inizialmente a 7 bit, in grado di codificare solo 128 caratteri, lo standard a 8 bit si è affermato via via tra la fine degli anni ’60 e l’inizio dei ’70). Secondo questo standard, ad esempio, la parola “FILOSOFIA” viene codificata in binario come segue:

01000110 01101001 01101100 01101111 01110011 01101111 01100110 01101001
01100001

Ad ogni lettera corrisponde un numero decimale stabilito dalla tabella ASCII. Ad esempio, la lettera *F* corrisponde al numero 70. Questo numero viene poi convertito in binario scomponendolo in potenze di 2 che, sommate tra loro, restituiscono esattamente 70: 64 + 4 +

Arithmetic, introdotto nel 1985 e aggiornato più volte (l’ultima revisione approvata è del 22 luglio 2019). Questo standard regola l’utilizzo dei bit e stabilisce le modalità di arrotondamento obbligatorie, ed è attualmente sotto revisione continua per adattarsi alle esigenze delle architetture moderne. Lo scopo dello standard è garantire che, dati gli stessi input, indipendentemente dalla realizzazione – hardware, software o combinata – il calcolo in *floating-point* produca risultati identici e gestisca gli errori in modo coerente (IEEE, Std 754-2019). In Goldberg [Goldberg, 1991], in riferimento agli standard *IEEE 754*, viene fornita una trattazione divenuta un classico sul *floating-point* e sui limiti della rappresentazione binaria dei dati, nonché sulle possibilità di errore nei sistemi digitali - arrotondamento, cancellazione, overflow - e su come il design di hardware, compilatori e linguaggi possa influenzarne l’impatto.

³¹⁷ The term is coined from bite, but respelled to avoid accidental mutation to bit

³¹⁸ “A sequence of eight bits is enough to encode the letters of the Roman alphabet in upper and lower case; the numerals, punctuation, and other symbols; and other special characters that may be used”

2. Si parte dalla potenza di 2 più alta contenuta in un byte, cioè 128, e si verifica se essa può essere “accesa” (cioè se non supera il numero da rappresentare). Nel caso di 70, 128 è troppo grande, quindi il primo bit resta spento (0). Solo i bit corrispondenti a 64, 4 e 2 vengono accesi (1), ottenendo così la sequenza binaria 01000110. Lo stesso procedimento si applica a ciascuna delle altre lettere della parola.

La scelta della codifica binaria, inoltre, presenta un’ulteriore caratteristica decisiva. Abbiamo visto come la logica booleana sia implementabile nei circuiti elettronici e come i neuroni possano essere considerati computazionalmente equivalenti a questi. In questo contesto, la notazione in base 2 si configura come la scelta più “naturale” per dare materialità a questa traiettoria teorica e renderla concretamente implementabile nel calcolatore. “I bit sono rappresentati con pari efficacia in termini semantici (significato vero/falso), logico-matematici (che sta per 1/0) e fisici (transistor = on/off; cancello = aperto/chiuso; circuito elettrico = basso/alto voltaggio; disco o nastro = magnetizzato/non magnetizzato; cd = presenza/assenza di “buchi” ecc.” [Floridi, 2010]. Si comprende allora come la notazione binaria, nella forma che ha dato corpo al paradigma digitale, sia divenuta la piattaforma comune di scienze cognitive, *computer science*, teorie dell’informazione e della computabilità. “La macchina calcolatrice dev’essere perciò una macchina logica, oltre che aritmetica, e deve combinare le alternative secondo un algoritmo sistematico” [Wiener, 1948, p. 162, trad. it.].

A questa equivalenza tra logica e computazione si aggiungono “i preconetti del fisiologo”, ovvero le preferenze per il carattere discreto, per il funzionamento “discontinuo del ‘tutto o niente’”, che aiutano a comprendere “perché la maggior parte delle ricerche compiute sulle analogie meccaniche del cervello siano state fatte su macchine che più o meno funzionano sul principio numerico” [Wiener, 1950, p. 91, trad. it.].

Abbiamo visto nel capitolo precedente come il calcolatore sia stato assunto, dalle scienze cognitive e dall’Intelligenza Artificiale “classica” - la cosiddetta GOFAI, di tipo simbolico - come metafora della mente. Sull’appropriatezza di questa metafora, dal punto di vista del calcolatore, ritorneremo in seguito.

I calcolatori, le macchine, “riconoscono bit riconoscono bit dal punto di vista fisico, operano logicamente sulla base di tale riconoscimento e, di conseguenza, manipolano dati con modalità che reputiamo dotate di significato” [Floridi, 2010]. Questa attribuzione di significato pone un’altra volta l’accento sull’interazione tra la computazione e le facoltà della mente e del cervello umani, già notata sin nell’analisi di nozioni fondamentali come quella di algoritmo. Ma

significa una cosa ulteriore. L'unica «intelligenza» ascrivibile alle macchine, ad oggi, è quella di saper distinguere tra alternative discrete binarie: “La sola scintilla di intelligenza che si può oggi accordare pacificamente a un computer riguarda la capacità dei suoi dispositivi e circuiti di discriminare tra dati binari. Se un computer percepisce qualcosa, non è altro che la differenza tra il basso o alto voltaggio al quale i suoi circuiti sono programmati per agire” [Floridi, 2010]³¹⁹.

Tab.4 I multipli del *byte*

Valore	Simbolo	Nome Italiano
10^3	kB	Kilobyte
10^6	MB	Megabyte
10^9	GB	Gigabyte
10^{12}	TB	Terabyte
10^{15}	PB	Petabyte
10^{18}	EB	Exabyte
10^{21}	ZB	Zettabyte
10^{24}	YB	Yottabyte
10^{27}	RB	Ronnabyte
10^{30}	QB	Quettabyte

I prefissi decimali del Sistema Internazionale delle unità di misura (SI), stabiliti dal *Bureau International des Poids et Mesures* (BIPM), sono stati introdotti progressivamente nel corso del tempo per rispondere all'esigenza di denominare quantità sempre maggiori. I prefissi fino a *tera-* (10^{12}) risalgono già alla fondazione ufficiale del SI nel 1960. Successivamente, *peta-* (10^{15}) ed *exa-* (10^{18}) sono stati aggiunti nel

³¹⁹ Si può notare come questo punto chiarisca ulteriormente la dicotomia *sintassi/semantica* nel dibattito sull'intelligenza dei calcolatori, posta al centro dell'attenzione dal celeberrimo esperimento mentale della *Stanza Cinese* [*Chinese Room*] di John Searle [Searle, 1980]. In esso, Searle immagina una persona chiusa in una stanza che manipola *simboli formalmente interpretati* in cinese seguendo regole sintattiche, senza però capire nulla del cinese stesso: analogamente, un computer può trattare simboli, ma non comprenderne il significato. Se, come afferma Searle, i computer possiedono solo una sintassi e non una semantica [In the linguistic jargon, they have only a syntax but no semantics; Searle, 1980], allora anche la capacità “protosemantica” della macchina – cioè la discriminazione tra alternative binarie, il fatto che al computer “suona strano nella quantità di bit che sta processando” e completare o integrare attraverso calcoli matematici se i dati sono persi o in completi [Floridi, 2010], uno degli altri grandi vantaggi del digitale rispetto all'analogico – indica non l'inesistenza, ma l'irrelevanza del punto di vista semantico [Floridi, 2005]. Tuttavia, questa capacità minima di riconoscimento (o di riconoscimento dell'assenza) è insufficiente per attribuire alle macchine un autentico punto di vista semantico, capace cioè di risolvere i paradossi e i problemi tipici dell'IA, come quello dell'*homunculus* o il problema del radicamento dei simboli. Come sottolinea Floridi, le attuali acquisizioni dell'IA [Intelligenza Artificiale] si basano su misure sintattiche, mentre l'IN [Intelligenza Naturale] si fonda su risorse semantiche [Floridi, 2005].

1975, mentre *zetta-* (10^{21}) e *yotta-* (10^{24}) sono stati introdotti nel 1991. Più recentemente, il 18 novembre 2022, durante la 27^a Conferenza Generale dei Pesi e delle Misure (CGPM), sono stati ufficialmente introdotti anche i prefissi *ronna-* (10^{27}) e *quetta-* (10^{30}). La classificazione completa si trova sul sito ufficiale: <https://www.bipm.org/en/measurement-units/si-prefixes>

Visti i vantaggi della computazione digitale e l'affermazione della notazione binaria, è ora necessario esaminare, in linea con i capisaldi concettuali dello sviluppo storico dei calcolatori, alcuni “luoghi genetici” nei quali si è affermata e progressivamente sviluppata l'implementazione del paradigma digitale, che darà luogo a macchine sempre più veloci e potenti. In particolare, sul piano dell'hardware, fu decisivo il passaggio dall'elettromeccanica all'elettronica: in primis, l'introduzione dei tubi a vuoto (noti nel Regno Unito come valvole termoioniche), che sostituirono i relè e resero possibile una netta accelerazione nei tempi di elaborazione e un sensibile incremento dell'affidabilità dei processi computazionali automatici.

Sviluppi in cui il pensiero di Alan Turing e John von Neumann, insieme ai contributi ingegneristici e concettuali di figure come Tommy Flowers, J. Presper Eckert e John Mauchly, inaugura - come in un racconto di Stanisław Lem - una genealogia che porterà alla nascita del calcolatore elettronico, all'elaborazione del concetto di macchina universale a programma memorizzato, l'*architettura di von Neumann*: un sistema in cui dati e istruzioni sono conservati nella stessa memoria e processati attraverso un'unità centrale di controllo e un'unità aritmetico-logica condivise. Una struttura destinata a diventare la base dell'organizzazione dei computer moderni.

Questo sviluppo fu accelerato dalle esigenze imposte dallo sforzo bellico della Seconda guerra mondiale: da un lato, la necessità di costruire macchine in grado di decifrare i messaggi cifrati tedeschi più velocemente delle Bombe inizialmente realizzate a Bletchley Park per contrastare le temibili Enigma; dall'altro, la crescente richiesta di potenza di calcolo per supportare i complessi modelli matematici impiegati nello sviluppo degli ordigni nucleari.

E più forti delle bombe erano, nella riflessione e negli interessi di John von Neumann, i pensieri intorno allo sviluppo del calcolatore. Come racconta il fisico Freeman Dyson, von Neumann - che aveva lavorato con grande entusiasmo a Los Alamos durante la Seconda guerra mondiale come consulente per la progettazione della bomba atomica - “anche allora capiva che l'energia nucleare non sarebbe stata il tema principale del futuro dell'umanità” [Dyson, 1979,

p. 226, trad. mia]. Nel 1946 il matematico di origine ungherese incontrò un suo vecchio amico, il fisico russo naturalizzato italiano Gleb Wataghin, che lo provocò con tono scherzoso: «Ciao, Johnny» disse Wataghin: «suppongo che ormai non ti interessi più la matematica. Ho sentito dire che adesso pensi solo alle bombe.» La risposta di von Neumann fu emblematica: «È del tutto sbagliato. Sto pensando a qualcosa di molto più importante delle bombe. Sto pensando ai calcolatori» [Dyson, 1979, p. 226, trad. mia]³²⁰.

Ora bisogna prendere in considerazione il materializzarsi storico di questo paradigma digitale – binario- nelle macchine,

³²⁰ “During the Second World War, von Neumann worked with great enthusiasm as a consultant to Los Alamos on the design of the atomic bomb. But even then he understood that nuclear energy was not the main theme in man’s future. In 1946 he happened to meet his old friend Gleb Wataghin, who had spent the war years in Brazil. “Hello, Johnny,” said Wataghin. “I suppose you are not interested in mathematics any more. I hear you are now thinking about nothing but bombs,” “That is quite wrong,” said von Neumann. “I am thinking about something much more important than bombs. I am thinking about computers.”

Cap. 4: Il mondo in un nastro: Macchine universali, calcolatori seriali

1. HQIBPEXEZMUG, o di come uno sconosciuto gentiluomo tedesco diede avvio alla Rivoluzione dei Computer

Come abbiamo indicato nelle pagine precedenti, il passaggio dalla carne all'hardware - dall'operatore impegnato in laboriosi calcoli con carta e matita alle stanze occupate da tonnellate di acciaio - si è sviluppato lungo alcuni vettori ben precisi: la digitalizzazione, la convergenza tecnologica, lo sviluppo dell'elettronica e la trasformazione delle modalità operative del calcolatore, dagli output cartacei alle prime visualizzazioni elettroniche. Tutti questi elementi hanno portato all'incarnarsi in hardware sia delle idee che sostenevano la metafora cibernetica, sia delle concezioni di Turing sull'universalità dei calcolatori.

Questa materializzazione in dispositivi concreti avvenne principalmente sulle due sponde dell'Atlantico, in un periodo storico ben preciso: la Seconda guerra mondiale, con le sue esigenze di elaborazione statistica, di procedure di decrittazione e di calcoli balistici (fino alla progettazione della bomba atomica). Ed è proprio da qui, sul versante inglese, a Bletchley Park, che prende avvio la nostra storia: un percorso che condurrà all'architettura di von Neumann e alla realizzazione concreta della macchina universale di Turing

Allo scoppio della guerra, la macchina da cifratura Enigma era già tecnologicamente obsoleta. Messa sul mercato nel 1923 e commercializzata in tutta Europa, la macchina era "lenta e scomoda da usare" [Copeland, 2006b, p. 36, trad. mia]³²¹. Richiedeva, inoltre, un numero spropositato di operatori: ben sei. Erano necessari un dattilografo, un assistente e un operatore radio a ciascun capo della trasmissione [Copeland, 2012, p. 116, trad. it.]. E anche quest'ultima era tutt'altro che efficiente.

L'operazione iniziava con un testo tedesco (il *clear*, o *plaintext*). Una volta cifrato, si otteneva il testo crittografato, il *ciphertext*. Il messaggio in chiaro era spesso mescolato con abbreviazioni o termini in gergo militare. L'addetto alla cifratura digitava il *plaintext* sulla tastiera dell'Enigma. A ogni tasto premuto, una lampadina illuminava una lettera. Digitando HITLER, potevano illuminarsi le lettere FLKPIM. Queste venivano annotate da un assistente. Al

³²¹ "The Enigma cipher machine was slow and cumbersome to use".

ciphertext venivano poi aggiunte delle informazioni che completavano il messaggio. Il tutto veniva quindi trasmesso via radio in codice Morse da un operatore.

All'altro capo, quello della ricezione, il processo avveniva all'inverso. Il marconista riceveva il messaggio Morse e lo consegnava all'addetto alla cifratura che, utilizzando una macchina Enigma impostata esattamente come quella del mittente, leggeva sul pannello le lettere in chiaro del messaggio originale [Copeland, 2006b, p.36].

Nel 1942 la macchina bellica nazista iniziò a utilizzare strumenti di cifratura che rappresentavano un *upgrade* rispetto ai dispositivi Enigma. A quanto si può leggere nel voluminoso rapporto tecnico *General Report on Tunny*, scritto nel 1945 e desecretato solo nel 2000, redatto da tre crittoanalisti di Bletchley, Jack Good (1916–2009), Donald Michie (1923–2007) e Geoffrey Timms (1903–1982) [Good, Michie, Timms, 1945]³²², le *Funktruppen*, le unità di segnalazione tedesche, furono dotate di nuovi dispositivi wireless che utilizzavano telescriventi per l'invio e la ricezione dei messaggi. Questi venivano cifrati e trasmessi via etere anziché via cavo. Tra la telescrivente mittente e il trasmettitore radio venivano interposte delle macchine di cifratura, le macchine SZ o *Schlüsselzusatzgeraet*. In esse la cifratura non veniva applicata direttamente al messaggio, ma ai segnali della telescrivente, e il testo cifrato non era mai visibile agli operatori. La ricezione avveniva su un nastro adesivo continuo, privo di simboli di formattazione (come il ritorno carrello e l'avanzamento riga) e del segnale acustico (il suono del campanello) [Good, Michie, Timms, 1945, 11A(c), p.3].

Questi dispositivi a telescrivente furono ribattezzati dal servizio crittoanalitico britannico con il nome in codice FISH. “I tedeschi si riferivano a uno dei loro collegamenti radio con il nome di *Sägefisch*. Forse il termine *Fish*, usato a Bletchley, deriva da questo” [Tunne, 2006, p. 355, trad. mia]³²³.

L'unità mobile FISH era composta da due camion: il *Betriebswagen*, con due macchine cifranti, telescriventi e perforatore a cinque impulsi; e il *Sendungswagen*, con il trasmettitore di telegrafia senza cavi (WT). Il ricevitore WT, trasportato dai due camion, era posizionato a circa mezzo miglio dal *Betriebswagen* e collegato via cavo. Nei centri stabili, come Berlino, o Parigi nel 1943, l'apparato era installato in stazioni fisse. Le tre nuove e temibili macchine *Pesce* note

³²² Il *General Report on Tunny* è disponibile sul sito *The Alan Turing Internet Scrapbook* all'indirizzo https://www.alanturing.net/turing_archive/archive/index/tunnyreportindex.html e, con ampia prefazione e note editoriali, è stato pubblicato nel volume *Breaking Teleprinter Ciphers at Bletchley Park*, a cura di James A. Reeds, Whitfield Diffie e J. V. Field (2015).

³²³ “The Germans referred to one of their radio links as ‘Sägefisch’. Perhaps Bletchley’s term ‘Fish’ derived from this”.

ai britannici erano tre: *Sturgeon*, utilizzata in particolar modo dalla *Luftwaffe*³²⁴, *Trasher* e *Tunny*. [Good, Michie, Timms, 1945, 11A(c), p.5].

Dopo un primo utilizzo sperimentale, attraverso il sistema di collegamento *Hellschreiber*, una rudimentale forma di fax avviato nel giugno 1941 su un collegamento tra Berlino e la Grecia, passando per Vienna, Atene e Salonicco [Copeland, 2006c, p. 87; Good, Michie, Timms, 1945 41A (a) p. 298], la macchina *Tunny* entrò in uso regolare a partire dall'autunno del 1942. Il collegamento unico iniziale fu sostituito da un sistema a due linee, battezzato dagli inglesi *Codfish* [merluzzo]; ad esso si affiancò un secondo canale, *Octopus* [polpo], che da Königsberg (la città prussiana del grande Kant, oggi l'exclave russa di Kaliningrad) arrivava fino all'Ucraina e alla Russia meridionale. L'attivazione di questo secondo collegamento segnò la fine della fase sperimentale e l'inizio dell'impiego preferenziale di *Tunny* da parte dell'Alto Comando tedesco, in sostituzione di Enigma, per la cifratura di comunicazioni della massima importanza strategica destinate ai comandanti dei gruppi d'armata sul campo [Copeland, 2012, p.117 trad. it; Copeland, 2004a, p.207; Good, Michie, Timms, 1945, 11D (b) (c), pp.13-14].

Il cuore dell'attacco di Bletchley Park a *Tunny* fu la costruzione del primo computer elettronico digitale *large-scale*: Colossus [Copeland, 2006a, p.1]³²⁵. Questa macchina rappresenta un passo cruciale nel percorso che ha portato alla computazione moderna nelle linee di sviluppo che abbiamo individuato.

Tunny era la macchina di cifratura prodotta dalla Lorenz modello *Schlüsselzusatz* (SZ)40. Misurava 48 cm in larghezza, 39 cm in profondità e 43 cm in altezza. Il suo cuore era costituito da 12 rotori, grazie ai quali "la cifratura e la decifratura erano completamente automatiche. Il testo cifrato trasmesso non veniva nemmeno visto dagli operatori tedeschi" [Copeland, 2006b, p. 37], che si trovavano ai due estremi delle due telescriventi connesse per la trasmissione dei messaggi.

La comunicazione tra le telescriventi avveniva mediante uno standard, il *codice Baudot-Murray*³²⁶, in cui ogni lettera o simbolo veniva tradotto in un codice binario a 5 bit (*pentabit*)

³²⁴ Un approfondimento su *Sturgeon* (lo storione), ovvero la macchina cifrante *Siemens & Halske T52*, con focus sulle sue caratteristiche tecniche, come il circuito interno di funzionamento *Pentagon*, alle sue debolezze strutturali e al suo impiego nel secondo dopoguerra (fino agli anni '60 in Europa occidentale e, con ogni probabilità, anche nella Germania Est tra i paesi del blocco sovietico), si trova in Weierud, 2006.

³²⁵ Central to the Bletchley attack on *Tunny* was Colossus, the world's first large-scale electronic digital computer.

³²⁶ Il codice Baudot, introdotto nel 1870 dall'ingegnere francese Jean Baudot, utilizzava sequenze di 5 impulsi binari per rappresentare lettere e simboli. La novità consisteva che tutti i simboli erano codificati da gruppi della stessa lunghezza. Nel 1901 l'ingegnere e inventore neozelandese Donald Murray ne sviluppò una versione

secondo un sistema internazionale. Ciascuno dei cinque impulsi era rappresentato da un segnale elettrico presente (*DOT*) o assente (*CROSS*), consentendo così la rappresentazione di 32 combinazioni distinte [Singh, 2006, p.16]. Questo sistema, non dissimile dal codice Morse, ma integralmente binario, costituì un precursore diretto della moderna codifica digitale – e può essere considerato ad essa formalmente corrispondente [Zabell, 2015, pp. LXXXII–LXXXIII].

Il *General Report on Tunny* descrive il funzionamento di questo sistema quando, come nel caso delle macchine crittografiche tedesche, due telescriventi sono messe in connessione:

Due telescriventi in comunicazione consistono in due macchine da scrivere elettromagnetiche ingrandite, collegate tramite cavo, e costruite in modo tale che qualunque cosa venga digitata su una delle tastiere venga stampata su entrambe le macchine. Quando un tasto viene premuto dall'operatore, la telescrivente invia lungo il cavo uno dei 32 segnali elettrici. Questi segnali consistono in cinque impulsi consecutivi, ciascuno dei quali può essere positivo (detto *DOT*) o negativo (detto *CROSS*), e attivano il tasto corrispondente della telescrivente ricevente. I 32 segnali sono noti come "LETTERS" e corrispondono ai tasti sulla tastiera del teleprinter [Good, Michie, Timms, 1945, 11A (a) p. 3; trad. it. mia]³²⁷.

L'operatore tedesco digitava un messaggio o inseriva un nastro nel telescrivente: il flusso, di natura binaria/digitale, di impulsi e non-impulsi attraversava la macchina *Tunny*, che lo mascherava con un flusso di caratteri oscuranti. "Questi caratteri oscuranti, noti come chiave, erano prodotti dai rotori della macchina Tunny." [Copeland, 2006b, p.39 trad. mia]³²⁸.

La cifratura avveniva in modo più sofisticato rispetto a quello dell'Enigma. Nella macchina Enigma, ogni ruota sostituiva una lettera con un'altra: per esempio, se la lettera 'N' entrava nella prima ruota, poteva uscirne una 'Z'; questa 'Z' veniva poi passata alla ruota successiva, dove poteva essere trasformata in una 'X', e così via. Le ruote della *Tunny* operavano invece in modo radicalmente diverso. Tutte le 12 agivano contemporaneamente per generare una singola lettera di chiave, la lettera cifrante, che veniva poi combinata con una lettera del

migliorata con caratteri di controllo per l'uso su telescriventi. Questa versione migliorata fu in seguito standardizzata tra il 1922 e il 1932 come *International Telegraph Alphabet No. 2* (ITA2).

³²⁷ "Two teleprinters in communication consist of two enlarged electromatic typewriters connected by cable, and constructed so that whatever is typed on either keyboard is printed on both type writers. When a key is depressed by the sender, the enlarged typewriter sends along the cable one of 32 electrical signals. These signals consist of five consecutive impulses, each of which may be positive (known as DOT) or negative (known as CROSS) and they operate the appropriate key of the receiving typewriter. The 32 signals are known as 'LETTERS' and correspond to the keys on the teleprinter keyboard".

³²⁸ "These obscuring characters, known as *key*, were produced by the Tunny machine's wheels".

plaintext. In questo modo, le lettere del messaggio venivano cifrate una a una [Cfr. Copeland, 2006b, p.40 trad. mia].

La cifratura procedeva allora in modo *discreto*, per ogni bit corrispondente ad una lettera e *pseudorandomizzato*, dovuto al movimento delle ruote che generavano le chiavi. “Per ogni lettera nel flusso di impulsi non cifrato, la macchina Tunny prende una lettera chiave (K) e la somma al testo in chiaro (P) per ottenere una lettera cifrata (Z)” [Good, Michie, Timms, 1945, 11B (b) p.6]³²⁹. La cifratura di Tunny avveniva prendendo la lettera in chiaro e “aggiungendole” - cioè combinandola, bit per bit, tramite un’operazione XOR (somma in modulo 2) sui cinque impulsi binari del codice Baudot corrispettivi a ciascun carattere - una lettera generata pseudocasualmente, per creare la lettera cifrata. [Singh, 2006, p.16; Good, Michie, Timms, 1945, 11B (a) p.6]. “In Tunny, $Z = P + K$, dove Z, P e K sono tutti ‘pentabit’, e l’addizione è eseguita modulo 2 in ciascuno dei cinque livelli o impulsi (cioè $0 + 0 = 0$, $0 + 1 = 1 + 0 = 1$, ma $1 + 1 = 0$)” [Good, 2006, p. 212 trad. mia]³³⁰.

Per quanto riguarda le ruote, nello specifico, la macchina Tunny ne usava 12 divise in gruppi: 5 ruote CHI (χ), che ruotavano sempre; 5 PSI (ψ), che ruotavano solo quando il motore (μ_1 e μ_2) lo permetteva; e 2 ruote motore.

Le CHI e le PSI generavano lettere da sommare al testo in chiaro per ottenere il testo cifrato. Il movimento irregolare delle PSI dipendeva da un segnale GO/STOP, controllato dalle ruote motore in combinazione con una limitazione, cioè una sorta di maschera fissa di controllo di movimento delle ruote derivata da altri pattern presenti sulla macchina (in particolare dalle ruote χ_4 e ψ_4), e non generata in modo indipendente. In breve, μ_1 ruotava sempre, mentre μ_2 si muoveva solo quando μ_1 mostrava una croce (GO). Il carattere attivo di μ_2 formava il *Motore Base* (BM), che, combinato con il carattere corrente della limitazione, determinava se il segnale risultava GO o STOP. Solo in caso di GO, le ruote ψ avanzavano; altrimenti restavano ferme e ripetevano il carattere di chiave precedente [Good, Michie, Timms, 1945, 11B (a) (b) (c) (d) (e) (f) (g) pp.6-9].

Le ruote erano dotate di camme disposte lungo la loro circonferenza. Queste, premendo un interruttore durante la rotazione, generavano impulsi elettrici. Il numero di camme determinava le posizioni della ruota. Le camme, regolabili dall’operatore, potevano essere disattivate: in tal caso, la ruota non generava più impulsi. Era proprio la disposizione

³²⁹ “For each letter in turn of the unciphered stream of impulse signals, the Tunny machine makes up a key-letter (K) and adds it to the plain text (P) to get a ciphered letter (Z).”

³³⁰ In Tunny, $Z = P + K$ where Z, P, and K are all ‘pentabits’ and the addition is modulo 2 in each of the five levels or impulses (i.e. $0 + 0 = 0$, $0 + 1 = 1 + 0 = 1$, but $1 + 1 = 0$)”.

attiva/inattiva delle camme a costituire il *pattern* della ruota. Queste liste, cambiate mensilmente prima dell'estate del 1944, e quotidianamente dal 1° agosto 1944, venivano fornite alle unità Tunny per configurare le macchine in modo identico da entrambi i lati della comunicazione [Cfr. Copeland, 2006b, pp.47-48].

Il *ciphertext* veniva così decifrato tramite un procedimento speculare: "Il messaggio cifrato era stato prodotto aggiungendo un flusso di chiave al testo in chiaro, quindi, aggiungendo esattamente le stesse lettere della chiave al testo cifrato, la macchina del destinatario annullava la cifratura, rivelando nuovamente il testo in chiaro"³³¹ [Cfr. Copeland, 2006b]. Ad esempio, supponendo che la parola in chiaro fosse COLOSSUS, il flusso di chiave cifrante aggiunto dalla Tunny del mittente potrebbe essere WZHI/NR9. Saranno proprio questi a essere aggiunti alla sequenza di lettere che compone COLOSSUS in questo modo:

C + W, O + Z, L + H, O + I, S + /, S + N, U + R, S + 9. Il risultato di questa cifratura è XDIVSDFE, ed è questa sequenza di lettere che viene inviata. La macchina del ricevente aggiunge la stessa chiave al messaggio cifrato, facendo emergere la parola originale: X + W, D + Z, I + H, V + I, S + /, D + N, F + R, E + 9, ossia COLOSSUS [Cfr. Copeland, 2006b, p.44].

A volte gli operatori tedeschi utilizzavano le stesse importazioni delle ruote, la stessa *chiave*, per due messaggi diversi. Queste circostanze erano definite *dephts* [Copeland, 2006b, p.50; Budiansky, 2006, p. 58]. Il 30 agosto 1941 fu un errore degli operatori tedeschi a fornire un aiuto insperato ai decrittatori di Bletchley Park. Questi avevano inviato due messaggi molto lunghi utilizzando lo stesso *indicatore*, il codice iniziale di 12 lettere, che comunicava come impostare le ruote di Tunny per la cifratura. L'indicatore in questione era il *depht*: "HQIBPEXEZMUG".

Quando questa stringa di caratteri fu analizzata, si scoprì che i due messaggi erano sostanzialmente identici, ma differivano per errori di battitura, spaziature e correzioni: in pratica, lo stesso testo era stato digitato a mano due volte. Il progressivo disallineamento tra le due versioni raggiungeva, alla 3976^a lettera - punto in cui il messaggio più breve terminava - uno scarto di oltre 100 caratteri [Good, Michie, Timms, 1945, 41C (b) p. 299]. Inoltre, nella seconda versione la parola *spruchnummer* ("numero del messaggio") era stata scritta in forma abbreviata, mentre nella prima era riportata per esteso.

³³¹ "This explains how the receiver's Tunny decrypted the ciphertext. The ciphertext was produced by adding a stream key to the plaintext, so by means of adding exactly the same letters of key to the *ciphertext*, the receiver's machine wiped away the encryption, exposing the plaintext again".

Questo *depth* si rivelò decisivo: rispetto agli altri intercettati dai britannici era molto più semplice da decifrare, poiché in ogni momento era possibile prevedere la lettera successiva del messaggio meno avanzato partendo da quello già parzialmente ricostruito. In questo modo si poté decifrare l'intero contenuto del messaggio più corto e risolvere l'ambiguità della chiave.

Questa violazione clamorosa dei protocolli di sicurezza crittografici fu un vero e proprio “dono del cielo” [*a gift from heaven*] [Budiansky, 2006, p. 56]. La decifrazione di questo *depth* fu affidata inizialmente al Tenente Colonnello John Tillman, “un relitto dei vecchi tempi, anteriori alla matematica e al calcolo automatico, in cui i codici e la loro decifrazione erano ancora prerogativa di metodi arcaici [Budiansky, 2006, p. 55 trad. mia]³³². Tuttavia la strada da seguire doveva procedere su un sentiero diverso dei metodi, “vecchia scuola dei decifраторi di codici, del tipo che avrebbe potuto uscire direttamente dalle pagine dei racconti di Edgar Allan Poe o Arthur Conan Doyle” [Budiansky, 2006, p.53 trad. mia]³³³. Il sentiero da seguire era quello dell'*analisi matematica*.

“Il fatto stesso che Tunny possa essere decifrata dipende dal fatto che P [il testo in chiaro], χ [lo *stream* generato dalle ruote chi], ψ [quello delle ruote psi], e quindi K [la chiave cifrante] e D [$K + \chi$, cioè il solo contributo psi isolato] *presentano caratteristiche statistiche, periodiche o linguistiche marcate, che li distinguono da sequenze casuali di lettere.*” [Good, Michie, Timms, 1945, 12B (c) p.17 trad. e enfasi mia]³³⁴.

L'attacco britannico a Tunny consisteva, spesso sulla base di sole analisi statistiche e linguistiche, nello sfruttare queste regolarità per scomporre i flussi in componenti elementari mediante la risoluzione di equazioni del tipo $K = \chi + \psi$.

Nei primi periodi, individuazione dell'indicatore – come HQIBPEXEMZMUG – permetteva di riconoscere con facilità i *dephts* e i *near-dephts*, cioè messaggi cifrati con le stesse o quasi le stesse impostazioni delle ruote. Se due messaggi avevano lo stesso *setting*, confrontandoli era possibile annullare la chiave di cifratura K e recuperare il *plaintext*, ricavando il pattern – la sequenza attiva/inattiva di camme – delle ruote. Questo in particolar modo quando il rapporto tra simboli *Baudot* DOTS e CROSSES era equilibrato, corrispondete a un rapporto $ab=1/2$ (con

³³² “A relic of the old pre mathematical, pre-computing days of codes and codebreaking”, Sull'episodio del 30 agosto 1941, noto come *The Tillman Break*, si veda Bauer, 2006, pp.370-377.

³³³ “Tiltman was definitely of the old school of codebreakers, the kind who could have walked off the pages of Edgar Allan Poe's or Arthur Conan Doyle's stories”.

³³⁴ “The fact that Tunny can be broken at all depends on the fact that P , χ , ψ , and hence K and D_i have marked statistical, periodic or linguistic characteristics which distinguish them from random sequences of letters”.

più punti • che croci ×), bastava una porzione abbastanza lunga di K per ricostruire il comportamento della ruota [Cfr. Good, Michie, Timms, 1945, 12B (c) (d) p.17].

Forti di questi risultati, i britannici, una volta decrittato il *depth* del 31 agosto - dopo un primo periodo frustrante, durato fino a novembre 1941 - riuscirono finalmente a decifrare i messaggi grazie a un sistema puramente speculativo, basato su carta e matita, elaborato dal giovane matematico di Cambridge Bill Tutte (1917-2002)³³⁵ reclutato a Bletchley Park. Senza aver mai visto una macchina Tunny prima della fine della guerra, gli analisti britannici riuscirono ad accedere all'intero traffico della comunicazione nazista. "A partire da luglio 1942, i primi messaggi tedeschi furono letti in tempo reale; da luglio a ottobre 1942, praticamente ogni messaggio fu decifrato" [Budiansky, 2006, p. 58, trad. mia]³³⁶.

Tuttavia, questo periodo durò poco. Ai successi estivi seguì, in termini di risultati una catastrofe autunnale. I tedeschi, resisi conto dell'intrinseca debolezza del sistema degli indicatori di 12 lettere utilizzati per il *setting* mensile delle ruote della Tunny, abbandonarono tale metodo e lo sostituirono con il sistema QEP.

Quest'ultimo serviva a sincronizzare le ruote della macchina Tunny tra mittente e destinatario attraverso l'utilizzo di appositi libricini [*QEP books*], ciascuno dei quali conteneva più di cento righe numerate (i numeri QEP), ognuna indicante dodici valori corrispondenti alle posizioni iniziali delle dodici ruote (le cinque χ , le cinque ψ e i due motori). Ogni libro aveva durata di un giorno, dopo di che veniva sostituito.

L'operatore dal alto del mittente selezionava una riga del libro, trasmetteva in chiaro "QEP xx" (le ultime due cifre del numero scelto), e entrambi gli operatori impostavano le ruote secondo le posizioni indicate. Poco prima dell'inizio della cifratura, il mittente trasmetteva il segnale "UMUM" e attivava la macchina, dando così avvio alla trasmissione cifrata. Dopo l'invio del messaggio, seguiva una verifica, manuale o automatica, della ricezione in maniera corretta, in base alla quale si apportavano delle eventuali correzioni. *Octopus* e *Codfish* funzionavano

³³⁵ William Tutte originariamente studente di dottorato in chimica a Cambridge si era successivamente invece appassionato alla matematica, "appariva, come lo stesso Turing, taciturno e assorto, intento a fissare a lungo un punto nel vuoto" [Cfr. Copeland, 2006b, p.51]. Tutte racconta in prima persona il metodo impiegato per analizzare la chiave di Tunny affidatagli dal capitano Gerry Morgan con le parole: "*See what you can do with this*" ["Vedi cosa riesci a fare con questo"]. Seguendo le tecniche apprese alla scuola di crittografia, iniziò studiando un solo impulso della chiave, trascrivendolo su griglie con vari periodi per rilevare pattern ricorrenti. Dopo tentativi con periodi 575 e 574, scoprì che con 41 colonne apparivano numerose ripetizioni verticali. Ne dedusse che il segnale fosse la somma modulo 2 di due sequenze: una periodica, generata da una ruota che avanzava sempre (χ), e una irregolare, generata da una ruota che talvolta restava ferma (ψ). Fu decisivo, come egli stesso ammette, "*a stroke of undeserved luck*" ["un colpo di fortuna immeritato"], l'errore dell'operatore tedesco che riuscì l'indicatore HQIBPEXEMUG. [Tutte, 2006, pp. 355-358].

³³⁶ "By July 1942, the first actual German messages were read currently; from July to October 1942, virtually every message was read".

secondo questo sistema. [Cfr. Copeland, 2006b, pp.45-46; Good, Michie, Timms, 1945, 11B (d) p.13].

Dopo questo cambio di metodologia, le speranze di decifrare i messaggi calarono vertiginosamente. Bletchley Park dovette tornare a confidare nei colpi di fortuna: l'invio, da parte dei tedeschi, di due messaggi nello stesso giorno con lo stesso indicatore di settaggio. I crittoanalisti britannici si ritrovarono così costretti a fare nuovamente affidamento sui *depths*, sulle coppie di messaggi cifrati con la stessa impostazione iniziale.

La fine dell'utilizzo degli indicatori all'inizio dei messaggi, paradossalmente, fu l'atto di avvio di un cambio di paradigma metodologico e scientifico dalla quale sarebbero sorte alcune macchine senza le quali lo sviluppo dei calcolatori così come li conosciamo non sarebbe stato possibile. Da quel giorno nero del novembre 1942, in cui i messaggi iniziarono a essere accompagnati solo da un semplice numero, che ovviamente i crittoanalisti britannici non possedevano, l'unica speranza di decifrazione rimaneva nei *depths*.

Eppure, questo episodio fu la scintilla di altro. Di una rivoluzione che prese la forma in una battaglia spietata tra macchine, quasi a prefigurare la nostra contemporaneità, quelle in cui la tecnologia non mette più l'essere umano con il suo ambiente, ma con un'altra tecnologia.

Tutte, uno dei protagonisti principali delle nuove metodologie con cui Bletchley Park si adeguò a combattere Tunny dopo l'adozione del sistema QEP, racconta, con una certa efficacia narrativa, il peso di questo evento nella storia che stiamo analizzando: "Suppongo che un ispettore tedesco abbia esaminato il processo di cifratura ed esclamato più o meno così: 'Ehi, state dando a quei bastardi informazioni che non avete bisogno di dargli! Non credo che gli siano servite a molto, ma è sbagliato in linea di principio. Basta con l'invio delle 12 lettere!'" Ma, prosegue Tutte, di fronte al disastro, i matematici britannici non si arresero: "Con una metafora spirituale, raccogliemmo le nostre matite e la carta a quadretti e uscimmo mesti dal Giardino dell'Eden. Fu in quel modo che, almeno secondo Bletchley Park, quello sconosciuto gentiluomo tedesco diede il via alla Rivoluzione dei Computer" [Tutte, 2006, pp. 362-363, trad. mia]³³⁷.

I *depths*, l'appiglio rimasto agli analisti britannici, potevano essere decifrati grazie a un metodo statistico manuale – "che richiedeva carta, matita e gomma" [Copeland, 2006e, p.381

³³⁷ "I suppose some German inspector had examined the process of encipherment and had exclaimed somewhat as follows: 'Hey, you're giving those bastards information that you don't need to give them! I don't suppose it has done them any good, but it's wrong in principle. Stop sending the 12 letters!' [...] Be that as it may, we had met with disaster. In spiritual metaphor we shouldered our pencils and squared paper and trudged glumly out of Eden. In some such manner did that unknown German gentleman, as judged at BP, set going the Computer Revolution".

trad. mia]³³⁸ – ideato da Alan Turing: la *Turingery*. Questo rimase l'unico metodo disponibile dopo l'introduzione del sistema QEP e l'abbandono degli indicatori.

La Turingery era un sistema di *wheel breaking* che permetteva di individuare i pattern delle camme sulle ruote, a partire da un segmento di chiave ricavato da un *depth*, isolando il contributo delle ruote χ e ricostruendone il pattern. Da questo era poi possibile dedurre anche i pattern delle ruote ψ e delle ruote motore.

Il concetto centrale di Turingery era l'introduzione dell'operatore Δ per individuare la "differenza" tra simboli consecutivi in una sequenza. "Il Turingery introdusse il principio secondo cui la differenza tra chiavi a distanza uno, oggi chiamata ΔK , poteva fornire informazioni non ricavabili dalla chiave ordinaria" [Good, Michie, Timms, 1945, 43B, p.298 trad. mia]³³⁹

Il Δ (delta) è il flusso di caratteri che si ottiene sommando ogni coppia di caratteri adiacenti nel flusso originale. L'idea centrale alla base dell'introduzione di questo operatore è che esso registra i cambiamenti nello stream operando su di essi mediante l'operazione XOR, l'addizione in modulo 2 applicata a ognuno dei simboli DOT o CROSS del codice Baudot. Se due simboli consecutivi sono uguali, si ottiene un punto ($\bullet$); se sono diversi, una croce ($\times$), corrispondenti rispettivamente a 0 e 1. In questo modo il delta evidenzia, lettera dopo lettera, dove il flusso resta costante e dove invece cambia [Cfr. Copeland, 2006d, pp.67, 68].

Applicando il Δ a una chiave, da questa era possibile dedurre informazioni sulla versione non deltaizzata [*delta-ed*]. Ogni impulso del flusso χ operava ciclicamente: il suo pattern si ripeteva dopo un numero di passi determinato dal numero di camme. La ruota χ_1 , avendo 41 camme, si ripeteva ogni 41 passi; la seconda ogni 31, e così via per le altre tre ruote, con 29, 26 e 23 camme. Una congettura corretta sull'identità del primo bit del primo impulso - cioè se fosse un punto ($\bullet$) o una croce ($\times$) - permetteva di risalire al valore del bit 41 posizioni più avanti. Supponendo di avere a disposizione x lettere, tale ipotesi consentiva di ricavare circa $x/41$ bit per la prima ruota, e rispettivamente $x/31$, $x/29$, $x/26$ e $x/23$ per le altre.

A partire da ciò, il metodo procedeva nel modo seguente. L'analista formulava un'ipotesi su un determinato punto della chiave deltaizzata, assumendo che le ruote ψ in quella posizione fossero rimaste ferme. Questa ipotesi aveva una probabilità del 50% di essere corretta. Le posizioni in cui le ruote ψ non si muovevano erano particolarmente interessanti, perché in quei

³³⁸ "Turingery was a hand method, involving paper, pencil, and eraser". Sulla recezione e l'uso della statistica a fini crittoanalitici a Bletchley Park si veda Zabell, 2015, pp.LXXV-CI.

³³⁹ "Turingery introduced the principle that key differenced at one, now called ΔK , could yield information unobtainable from ordinary key".

punti la chiave deltaizzata e la χ deltaizzata coincidevano. Se l'ipotesi era corretta, dato che la chiave era nota, la lettera di $\Delta\chi$ in quella posizione diventava anch'essa nota. A partire da questa si potevano ricostruire gli impulsi che compongono il pentabit, propagandosi avanti e indietro secondo i periodi delle rispettive ruote.

Avendo alcuni bit noti di $\Delta\chi$, li si poteva confrontare con la chiave cifrata ΔK per formulare nuove ipotesi sull'identità di altre lettere. Se ψ era ferma anche in quei punti, i nuovi bit potevano essere scritti e propagati seguendo il ciclo del pattern, completando così ulteriori porzioni della sequenza $\Delta\chi$. Si cercava così, progressivamente, di completare l'intero messaggio [Cfr. Copeland, 2006e, pp. 380-382; Good, Michie, Timms, 43B, pp.314-316].

La Turingery, "un metodo che richiedeva grande maestria [*artistry*]" [Tutte, 2000, p.14 trad. mia]³⁴⁰, si basava sulla fortuna, sulla pazienza e sulle abilità dell'analista. Inoltre, era fondata esclusivamente sull'uso dei *dephts*, una soluzione che non poteva soddisfare le esigenze della crittoanalisi britannica. "Fu allora che Tutte fece una delle scoperte matematiche più cruciali dell'intero progetto" [Budiansky, 2006, p.58 trad. mia]³⁴¹.

A partire dalle intuizioni della Turingery, in particolare dall'uso dell'operatore delta, Tutte propose un metodo di breaking basato esclusivamente su metodi statistici, senza l'uso di depths, l'"1+2 break-in" [Good, Michie, Timms, 15A (a), p.33].

Tutte si rese conto che due simboli consecutivi in una stringa Baudot tendevano a essere uguali con una frequenza maggiore rispetto a quella per cui differivano. Questo accadeva per l'effetto delle ruote ψ che spesso erano ferme tra un carattere all'altro. L'operatore Δ poteva allora essere usato per cogliere questa regolarità.

Sommando una stringa di simboli cifrati a sé stessa, traslata di una posizione [*shift-and-add*] il risultato conteneva il 70% di DOT se le ruote PSI avevano contribuito in modo determinante alla codifica. A partire dalle caratteristiche della lingua tedesca, ad esempio la conformazione dei dittonghi, il testo presentava, nei primi due bit, un comportamento simile.. Il *depht* e l'effetto combinato del testo in chiaro e delle ruote ψ poteva essere eliminato, lasciando una *traccia statistica* predominante del solo contributo delle ruote χ . Il metodo si sviluppava prendendo due impulsi consecutivi del messaggio cifrato C , C_1 e C_2 , e applicando l'operatore Δ alle rispettive equazioni $\Delta C_1 = \Delta P_1 + \Delta\chi_1 + \Delta\psi_1$ e $\Delta C_2 = \Delta P_2 + \Delta\chi_2 + \Delta\psi_2$.

Poiché le componenti $\Delta\psi_1$ e $\Delta\psi_2$ tendevano ad assomigliarsi, era possibile attenuarne l'effetto sommando le due espressioni. Si otteneva così $\Delta(C_1 + C_2)$, che si confrontava con la

³⁴⁰ "It is a method requiring great artistry".

³⁴¹ "It was then that Tutte made one of the most crucial mathematical discoveries in the project".

sequenza nota $\Delta(\chi_1 + \chi_2)$, la quale aveva un periodo di dato da quelli 41 e 31 delle due ruote corrispondenti.

Facendo scorrere, *a mano*, tutti gli allineamenti $\Delta(\chi_1 + \chi_2)$ su $\Delta(C_1 + C_2)$ in tutte le posizioni possibili e contando le concordanze, si poteva identificare l'allineamento corretto e, con esso, la posizione iniziale delle ruote χ_1 e χ_2 , ossia la configurazione iniziale. [Cfr. Tutte, 2006, p.365, Budiansky, 2006, pp.58-59].

“Ecco un metodo di *setting* delle ruote!” [Tutte, 2006, p.364 trad. mia]³⁴².

Il metodo statistico di Tutte andava però generalizzato. “Si poteva forse perfezionare fino al punto da trovare pattern ignoti delle ruote, invece di limitarsi a fissare ruote di cui si conosceva già il pattern?” [Tutte 2006, p.365 trad. mia]³⁴³. Nell'eventualità che l'estensione del metodo fosse possibile c'era tuttavia bisogno di un brano in tedesco molto lungo. “Proprio lì”, racconta Tutte “sul mio tavolo da lavoro, c'era un messaggio cifrato lungo circa 15.000 lettere” [Tutte, 2006, p.365 trad. mia]³⁴⁴.

Il matematico di Bletchley Park iniziò quindi a marcare il messaggio di 15mila lettere in blocchi 20 lettere ciascuno, con estrema attenzione. Era indispensabile non perdere l'allineamento: confondere, ad esempio, la lettera 11.614 con la 11.615 avrebbe compromesso l'affidabilità dell'intero procedimento. Doveva poi assumere, come nei casi precedenti, che la sequenza $\Delta(C_1 + C_2)$ fosse una versione sufficientemente vicina, seppur degradata della sequenza $\Delta(\chi_1 + \chi_2)$, sperando in una concordanza almeno del 55%. Sapeva che $\Delta(\chi_1 + \chi_2)$ era il risultato della somma di due sequenze periodiche, $\Delta\chi_1$ e $\Delta\chi_2$, con periodi rispettivi di 41 e 31.

Dopo di ciò, proseguì scomponendo la sequenza stimata nelle due componenti $\Delta\chi_1$ e $\Delta\chi_2$. Procedendo iterativamente, prese le mosse da una riga del rettangolo in cui la distribuzione dei simboli sembrava affidabile e la considerò come prima approssimazione di $\Delta\chi_1$. In base alla concordanza di DOT o CROSS, confrontò questa riga con le altre per stimare $\Delta\chi_2$. Ripetendo con pazienza e attenzione a mano il processo, alternando righe e colonne, Tutte arrivò a ricostruzioni che gli apparivano attendibili.

I risultati vennero poi affinati grazie a due osservazioni: le sequenze $\Delta\chi$ dovevano contenere un numero pari di croci, per via della loro natura periodica, e i pattern dovevano essere plausibili nel contesto della cifratura. Così ottenne stime affidabili, che non dovette più

³⁴² “Here was a method of wheel setting! “

³⁴³ “I began to dream of generalising the statistical method. Could it be improved so far as to find unknown wheel patterns instead of merely setting known ones?”

³⁴⁴ “There on my work table was a cipher message about 15,000 letters long”.

correggere, sia per le sequenze delta sia per le χ originali. Dopo la stima efficace delle prime due ruote, Tutte passò ad altre coppie di impulsi per determinare i pattern rimanenti delle ruote χ .

Malgrado temesse che per altre coppie di impulsi la corrispondenza tra il delta della sequenza cifrata e quello della ruota potesse essere meno precisa, poteva partire dal vantaggio di considerare noti uno dei termini della coppia $\Delta\chi$ come noto in base al passaggio precedente del suo metodo. Riuscì così ad ottenere i cinque pattern restanti delle ruote, χ . Questa chiave era a disposizione. Procedette alla rimozione di questa dal messaggio. Aveva ottenuto il messaggio in chiaro + ψ [Cfr. Tutte, 2006, p.365].

“Restava la grande operazione di de-psiing!” [Tutte, 2006, p.365]³⁴⁵.

Sebbene partisse con lo svantaggio manifesto, rispetto a chi settava le ruote, di non conoscere i pattern di PSI, Tutte fece comunque dei progressi. Pur avvantaggiato dalle caratteristiche del messaggio – strutturato in blocchi brevi, ricco di punteggiatura e di spazi consecutivi – riuscì a separare P da ψ analizzando il comportamento delle ruote PSI. Cercò di ricostruire la chiave motore osservando gli STOP e i GO delle ψ , confidando nel fatto che i loro movimenti seguissero un periodo di 2257, ricavato dal prodotto di quelli delle ruote motore di 61 e 37. Se una stessa lettera ψ si ripeteva dopo 2257 posizioni, le possibili triplette di testo in chiaro erano solo 32, e una sola quella plausibile³⁴⁶. Tutte ottenne le informazioni necessarie al recupero del messaggio grazie a questo metodo, il cosiddetto *metodo del rettangolo* [*rectangle method*] [Cfr. Tutte, 2006, pp. 366-367].

Il metodo “1+2 break in” e quello del rettangolo, sebbene avessero prodotto un successo della decrittazione dei messaggi Tunny, tuttavia erano lenti e noiosi. “È chiaro che il metodo del rettangolo raffinato sarebbe stato troppo tedioso per essere eseguito abitualmente a mano” [Tutte, 2006, p.367 trad. mia]³⁴⁷. Risultava, nella pratica, impossibile svolgere manualmente - con matita, carta, fortuna e pazienza - le procedure richieste, calcolare migliaia di combinazioni ed eseguire le complesse operazioni statistiche necessarie per decifrare ogni singolo messaggio. Questo avveniva, per giunta, in un momento in cui “i tedeschi trasmettevano ormai centinaia di messaggi Tunny al giorno”. [Budiansky, 2006, p.59 trad. mia]³⁴⁸.

Era necessario velocizzare i processi di decifrazione statistica elaborati da Tutte e ispirati da Turing dandogli *corpo* con metallo e circuiti: “Ciò che serviva era una

³⁴⁵ “Now for a grand operation of de-psiing!”.

³⁴⁶ Tutte nella sua ricostruzione rimane cauto riguardo al suo effettivo uso a Bletchley Park di questo metodo nella operazione “de-psiing”.

³⁴⁷ “Clearly the refined rectangle method would be too tedious to be done routinely by hand”.

³⁴⁸ “The Germans were by now transmitting hundreds of Tunny messages a day”.

meccanizzazione su una scala che nessuno aveva mai osato immaginare prima” [Budiansky, 2006, p.59 trad. mia]³⁴⁹. Qui si manifesta un passaggio che abbiamo già incontrato a proposito di Wiener e Bush: una complessa operazione statistico-computazionale che le circostanze non potevano più affidare alla sola risoluzione umana. Era necessaria una *macchina*.

Bletchley Park aveva cominciato a riempirsi, oltre che di militari e di donne³⁵⁰, anche di un ulteriore genere di personaggi, quelli del “*professor type*”, così definiti dal capo del GC & GS, il comandante Alastair Denniston (1881–1961) [Budiansky, 2006, p.57]. Tra questi ultimi c’era uno dei più “singolari” [odd] della già di per sé singolare compagine di uomini e donne che lavoravano al servizio crittografico, l’ex professore di matematica di Turing a Cambridge - l’uomo che lo iniziò ai problemi dei fondamenti della matematica legati teoria di Gödel - Max Newman. [Cfr. Budiansky, 2006, p.59].

Fu lui, “teoretico tra i teoretici”, che propose l’idea di meccanizzare il problema di Tunny [Budiansky, 2006, p.62]³⁵¹. Il comando di Bletchey accolse la sua richiesta e nel “dicembre del 1942 gli fu affidato il compito di sviluppare metodi meccanizzati per l’impostazione di Tunny”. [Good, Michie, Timms, 1945, 15A (b), p.3 trad. mia]³⁵².

Fu concesso a Newman di creare una propria sezione autonoma con il compito specifico di meccanizzare il *breaking* di Tunny. Questa sezione prese il nome di *Newmanry*, il cui staff

³⁴⁹ “What would be required was mechanisation on a scale that no one had ever dared dream of before”.

³⁵⁰ Sin dal 1937, con l’aggravarsi della tensione internazionale, le donne iniziarono a essere reclutate in massa per lavorare a Bletchley Park. Durante il picco della guerra, e con l’aumentare delle esigenze di forza lavoro in seguito alla meccanizzazione dei processi di decrittazione, come notato sopra, la popolazione femminile raggiunse le 7.500 unità, pari al 75% dello staff complessivo. Provenivano in gran parte dalla media borghesia ed erano inquadrare nei principali corpi ausiliari femminili: WRNS (*Women’s Royal Naval Service*), WAAF (*Women’s Auxiliary Air Force*), ATS (*Auxiliary Territorial Service*) e Land Army. Molte erano giovani e ben istruite: spesso diplomate o con esperienza scolastica superiore, a volte universitaria, erano selezionate, con stie tipicamente *british*, anche per la loro affidabilità sociale oltre che intellettuale. Impiegate in ruoli tecnici fondamentali, soprattutto nell’operazione dei macchinari come *Heath Robinson* e *Colossus*, le WREN, acronimo che indicava le donne del servizio navale, svolsero un ruolo compito imprescindibile nel processo di decrittazione. Ne è prova quello che si può leggere nel *General Report on Tunny*, dove si sottolinea tra le lezioni principali apprese dall’implementazione di Heath Robinson, proprio l’“abilità delle Wren nel calcolo” [“The ability of Wrens to compute” [Good, Michie, Timms 23Z, p.108]. Non a caso, Eleanor Ireland, WREN operatrice del *Colossus*, ha intitolato con orgoglio la sua memoria sul suo servizio a Bletchley “*We were the world’s first computer operators*”. In essa ricorda non solo l’addestramento tecnico e l’insistenza assoluta sulla segretezza, ma anche le condizioni di vita nel compound, il vitto e l’alloggio spartani, e i piccoli riti e le attività che scandivano la giornata quotidiana di una Wren a Bletchley [Ireland, 2017, pp.161-166]. Per un’altra testimonianza in prima persona, quella di Joan Valentine, che nel 2011 conobbe la Regina in visita a Bletchley, si veda Copeland (with Coughy and Valentine), 2017, pp.125-127. Si rimanda anche sulle WREN e sulla popolazione di Bletchley Park a Smith, 2015, oltre al già citato Hill, 2004. Ulteriori testimonianze in prima persona si trovano in Copleand et al., 2006, pp.159-167.

³⁵¹ “It was odd that it was Newman, the theoretician’s theoretician, who proposed mechanising the Tunny problem”.

³⁵² “The idea of using electronic counters to carry out these processes at a practically useful speed a was put forward by M.H.A. Newman and in December, 1942 he was given the task of developing machine methods of setting TUNNY”.

iniziale era composto da due ingegneri, 16 Wrens e due crittoanalisti, lo stesso Newman e Donald Michie (1923-2007). Quest'ultimo sembrava un candidato ancor meno plausibile del professore di matematica di Cambridge a cui doveva fare da assistente. Michie, infatti, era entrato nella Newmanry appena diciottenne, privo di qualunque preparazione tecnica o matematica. Aveva vinto una borsa di studio per studiare lettere classiche al Balliol College di Oxford, ma si era arruolato nella speranza di contribuire allo sforzo bellico e, al tempo stesso, ottenere un posto all'università una volta terminata la guerra. A tal fine, aveva frequentato un corso intensivo di lingua giapponese, di cui era venuto a conoscenza per sentito dire da un amico di famiglia impiegato al War Office.

Michie - che dopo la guerra divenne prima genetista e poi pioniere dell'intelligenza artificiale - fu uno dei tre autori del *General Report on Tunny*, insieme a Jack Good (1916-2009), giovane matematico di Cambridge e principale statistico della Newmanry, che proseguì poi la carriera accademica prima a Manchester, presso il Computing Machine Laboratory con Newman, e successivamente negli Stati Uniti, e a Geoffrey Timms (1903-1982), matematico accademico entrato a Bletchley Park nel 1944 e rimasto in servizio presso il GC & CS anche dopo la fine del conflitto³⁵³.

Il ristretto gruppo iniziale della *Newmanry* si ritrovò a lavorare sulla macchina nell'Hut 11 di Bletchley, una baracca a due stanze. I progetti e il design del macchinario erano stati affidati ad un altro *don* di Cambridge, ora in servizio presso il *Telecommunications Research Establishment* (TRE), C.E. Wynn-Williams (1903-1979). Il modello pilota fu commissionato nel gennaio del 1943 e iniziò a lavorare il giugno dello stesso anno [Copeland, 2006d, p.64; Michie, Good, Timms 15A (c), p.33].

Il collaudo del prototipo non fu dei più felici. Dalla macchina fuoriuscì una nuvola di fumo. Si era bruciata per sovraccarico una resistenza di grosse dimensioni. Le tavole di legno di cui era fatta la Baracca 11 furono in salvo per l'intervento di Harry Fensom (1921-2010), un ingegnere assegnato alla Newmanry, che bypassò il componente malfunzionante [Fensom, 2006, p.298].

³⁵³ È possibile trovare il punto di vista di Good e Michie sul loro lavoro alla *Newmanry* intorno alla meccanizzazione della decifrazione e agli sviluppi di questa che portarono a Colossus, in Good, 2006, pp.204-222; Michie, 2006, pp.223-246. Sulla Newmanry si rimanda a Copeland et al., 2006, pp.157-175. Il *General Report on Tunny* che prima di essere desecretato nel 2000 era noto come *The History of the Newmanry* or *The Newmanry Report* costituisce ovviamente una fonte storica, oltre che tecnica, preziosissima sul lavoro svolto.

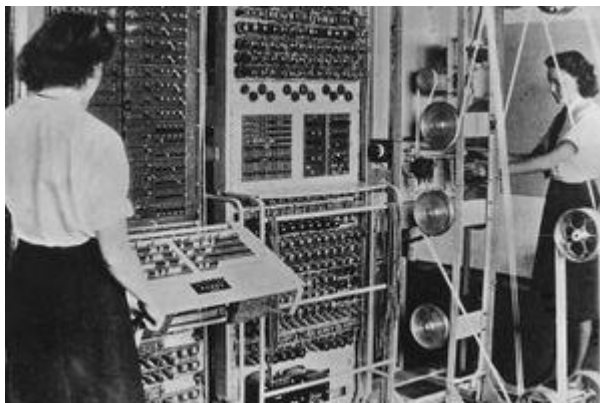
Le Wren di turno quel giorno ribattezzarono il dispositivo “*Heath Robinson*”, dal nome di un vignettista famoso all’epoca per i suoi disegni umoristici di complicatissimi congegni meccanici [Copeland, 2006d, p.65].

Heath Robinson [fig. 15] era composto da “un enorme telaio metallico [*frame*] in angolari di ferro, che ricordava un letto antiquato messo in verticale; divenne rapidamente noto come il *bedstead*” [Copeland, 2006d, p. 65, trad. mia].

Attorno al *bedstead* si trovavano due *loops* rotanti di nastro, sostenuti da un sistema di pulegge e ruote di legno del diametro 10 pollici (25 centimetri). I nastri da telescrivente (*teleprinter tape*) erano realizzati incollando tra loro le estremità di un nastro. I due anelli di nastro venivano azionati da ruote dentate che si agganciavano a una fila continua di fori presente sulla parte centrale di ciascun nastro. Uno stesso albero motore era responsabile del movimento dei due nastri, che scorrevano in modo sincronizzato. [Cfr. Copeland, 2006, p.65].

Questi due nastri che ruotavano attorno al *bedstead* erano il “*message-tape*” e il “*chi-tape*” [Copeland, 2006d, p.68]. Il nastro del messaggio presentava, alla fine, una sezione bianca (priva di dati), utile affinché la macchina potesse completare le operazioni logiche previste dalla routine (in termini contemporanei, il *programma*). Il secondo nastro conteneva i pattern delle ruote, cioè la sequenza delle camme attive/inattive. La lunghezza dei nastri era variabile - dipendeva dal messaggio da analizzare - ma con un accorgimento: la lunghezza del nastro del *chi-tape* veniva scelta in modo da non avere fattori comuni con quella del nastro del messaggio. In questo modo, a ogni rivoluzione completa del messaggio, i pattern delle ruote riprendevano sempre da una posizione diversa rispetto al testo cifrato. [Cfr. Fensom, 2006, pp.298-299].

Fig. 15. Heath Robinson e le sue operatrici WREN alla Newmanry [da Encyclopaedia Britannica. *Heath Robinson*]



Come deciso nelle discussioni preliminari alla costruzione della macchina i pattern di fori presenti sui nastri erano letti fotoelettricamente. Venivano impiegate due serie da dieci celle fotoelettriche – inizialmente a gas, poi sostituite con quelle a vuoto ad alta velocità – per leggere i fori dei nastri *message* e *chi*. Altre tre erano impiegate per i fori di trascinamento, inizio e fine [Fensom, 2006, p.298; Michie, Good, Timms 15A (b), p.33]³⁵⁴.

Una volta letti fotoelettricamente, i pattern fori/non-fori sui nastri convertiti in impulsi elettrici. Questi poi erano modificati e combinati dall'unità di combinazione [*combining unit*] (una sorta di unità logica). Qui si performavano le operazioni *XOR* sugli impulsi. Input e output delle porte logiche simili ai cavi a doppio connettore di un centralino. Questa unità era il fulcro della macchina. Le diverse combinazioni di cavi e prese, eseguite a mano, spesso dalle Wrens, rappresentavano la modalità, primitiva e rudimentale di programmazione di Heath Robinson [Cfr. Copeland, 2006d, p.65; Fensom, 2006, p.298; Good, Michie Timms, 1945, 54A-G, pp. 345-362]

Il numero di impulsi risultante dalle operazioni veniva poi computato dall'unità di conteggio [*counting unit*] di Wynn-Williams. [Copeland, 2006, p 65]. Questi, sulla base della sua esperienza credeva nell'impiego del minor numero possibile di valvole [Fensom, 2006, p.299]³⁵⁵. I modelli Heath Robinson, quindi, poggiavano fortemente sull'uso di componenti elettromeccaniche, “effettuavano il conteggio utilizzando quattro stadi elettronici in base 2, seguiti da quattro stadi meccanici a relè in base 5” [Good, Michie Timms, 1945, 51 (e), p.326]³⁵⁶.

La velocità dei nastri di Heath Robinson era di 2000 caratteri al secondo - i caratteri erano le alternative DOT/CROSS, corrispondenti agli 0 e a gli 1 binari, rappresentati dai fori si di trascinamento [Copeland, 2006d, p.65; Diffie, 2015, p.lii]. La quantità di caratteri attraversati in un ciclo equivaleva al prodotto delle lunghezze delle ruote moltiplicata per la lunghezza del messaggio. “A una velocità di 2000 caratteri al secondo, un test delle ruote χ_1 e χ_2 (lunghe rispettivamente 41 e 31 posizioni) su un messaggio di circa 2000 caratteri avrebbe richiesto approssimativamente mezz'ora” [Fensom, 2006, p.298]³⁵⁷

³⁵⁴ “A number of schemes were considered, including that of sliding photographic plates over E.1 each other, a method later perfected in U.S. It was soon settled that the best machine for the early experimental stages was one which read a ‘message-tape’ and a ‘wheel-tape’ photo-electrically, and combined them electrically before counting”.

³⁵⁵ “Wynn-Williams believed in using the minimum number of valves, because his experience was with equipment for mobile use”.

³⁵⁶ “The earlier Robinsons counted in four electronic scales of 2, followed by four mechanical relay scales of 5”.

³⁵⁷ “At 2000 characters per second, a test of χ_1 and χ_2 (of lengths 41 and 31) against a message of about 2000 characters would take approximately half an hour”.

Possiamo vedere come nell'architettura e nella realizzazione della macchina i metodi statistici impiegati da Tutte, e ispirati da Turing, acquistino materialità nell'automazione. I due nastri, il *message-tape* e il *chi-tape* rappresentavano il messaggio cifrato e quello contenente le ipotesi di *chi-stream*. Le operazioni di confronto e di allineamento, che prima dovevano essere pazientemente eseguite a mano, ora scorrevano su nastro. "L'unità di combinazione eseguiva l'operazione di somma e quella di delta, mentre i contatori elettronici di Wynn-Williams producevano i punteggi" [Copeland, 2006d, p.69 trad. mia]³⁵⁸.

Un confronto sistematico tra il messaggio e i possibili allineamenti delle ruote χ_1 e χ_2 , su una lunghezza di 10.000 caratteri, comportava l'esecuzione e il test di oltre 12 milioni di operazioni (in numero di caratteri andava moltiplicato per il prodotto del numero di camme delle due ruote). Utilizzando un lettore di nastro standard dell'epoca, in grado di leggere appena 10 caratteri al secondo, tale procedura avrebbe richiesto circa due settimane di lavoro continuo. Heath Robinson, con la sua velocità di lettura di 2000 caratteri al secondo, permetteva di completare lo stesso ciclo in poco più di due ore. [Cfr. Fensom, 2006, p.299-300].

Il dispositivo funzionava. Dimostrava nella pratica l'efficacia del metodo di Tutte e la validità dell'utilizzo delle macchine. Ma, al contempo, "soffriva di handicap intollerabili" [Copeland, 2006, p.71 trad. mia]³⁵⁹. Il *General Report on Tunny* elencava una impietosa lista di questi *handicap* [Cfr. Good, Michie, Timms, 1945, 52b (b) p.328]³⁶⁰.

Heath Robinson, nonostante i miglioramenti in termini di velocità, impiegava ancora diverse ore per decifrare un singolo messaggio e non risultava abbastanza rapido per le esigenze pressanti della guerra. Inoltre, non era affidabile: a parità di input, i contatori

³⁵⁸ "The combining unit did the adding and the delta-ing, and Wynn-Williams' electronic counters produced the scores".

³⁵⁹ "Heath Robinson suffered from 'intolerable handicaps'."

³⁶⁰ "(i) There was at first no printer: the operators (two in number) had to write down the fleeting figures on display: this was a fruitful source of error. (ii) The distance between the gate where the tape was scanned and the sprocket-wheel which drove it was six inches, so that the stretching of tapes alone was sufficient to put tapes out of alignment. (iii) The position counter recorded, not the relative position of the two tapes, but the number of revolutions completed: from this the relative position can be found but with great risk of erroneous calculation. (iv) Heath Robinson would not tolerate long stretches of dots or of crosses, so that elaborate tapes, with additional opportunities for making mistakes had to be devised to avoid this. (v) The minimum text length was 2000. If it was less, rubbish had to be inserted in such a way that it was not counted. (vi) There was no spanning. (vii) The forms of impossible conditions were severely limited. (viii) The counters were only partly electronic. (ix) At first Heath Robinson was unable to obtain results, even if not itself at fault, because the tapes, not being subject to a proper system of checks, were incorrectly made." A questa lista vanno aggiunte 29 difficoltà individuate dagli autori del *General Report*, che andavano dalle semplici disattenzioni o errori casuali degli operatori, a problemi strutturali dell'hardware, alla tendenza ai picchi di errore delle componenti meccaniche come i e guasti ai relè o nastri danneggiati, passando per l'interpretazione incerta dei dati, la scarsità di materiali e strumenti, fino alle condizioni di lavoro insalubri e intollerabili che causavano malattie allo staff [Cfr. Good, Michie, Timms.1945, 23Z, pp.110-111].

restituivano talvolta risultati differenti. A questi limiti si aggiungevano errori “umani” - quasi strutturali all’impiego dei nastri, come punzonature errate o non rilevate (soprattutto nei *chi-tape* molto lunghi), che potevano persino bloccare del tutto la macchina - e l’usura fisica dei nastri e dei fori che indicavano i pattern. Ma il problema più grave era la frequente perdita di sincronia tra i due nastri: come già accadeva nel metodo “manuale” di Tutte, il disallineamento comprometteva irrimediabilmente l’intero calcolo. [Cfr. Copeland, 2006d, p.71].

William Newman, figlio di Max, scrive che il padre “rimaneva alla Newmanry dalle 9 del mattino fino a mezzanotte per settimane intere. Lui e il suo piccolo team, sotto enorme pressione per ottenere risultati, si confrontavano continuamente con la complessità, l’inaffidabilità e la lentezza di Heath Robinson. [Newman, 2006, p.182-183 trad. mia]³⁶¹

Malgrado questi difetti, “La flessibilità di Heath Robinson, utile a fini sperimentali, rese facile individuare i requisiti essenziali di una macchina destinata a decifrare Tunny” [Good, Michie Timms, 52 (e), p.329 trad.mia]³⁶².

“La domanda ora era: come costruire una macchina migliore - una domanda per un ingegnere. Con un colpo di genio, l’esperto di elettronica Thomas H. Flowers risolse tutti questi problemi” [Copeland, 2006d, p.71 trad. mia]³⁶³

Se “la fabbrica di mostruosi calcolatori elettronici di Newman, dedicata alla decifrazione di Tunny, offriva uno scorcio del futuro” [Copeland, 2006d, p.63 trad. mia]³⁶⁴, in questo spiraglio iniziava a intravedersi la sagoma di una figura *colossale*.

La macchina di Flowers, insieme a quelle che si andavano elaborando e costruendo sull’altra sponda dell’Atlantico, rappresenteranno dei passi decisivi sulla via del calcolatore moderno e per l’impatto del digitale sulla nostra contemporaneità.

2. Colossi, valvole e profeti: le vie del digitale

Thomas Harold “Tommy” (o come preferì in tarda età, *Tom*) Flowers (1905–1998) non aveva mai amato i relè. Nipote di un cacciatore di frodo, era nato e cresciuto tra i quartieri

³⁶¹ In 1943 Max was at the Newmanry from 9 a.m. to midnight for weeks on end. He and his small team, under great pressure to deliver results, grappled constantly with Heath Robinson’s complexity, unreliability, and lack of speed.

³⁶² The flexibility of Heath Robinson for experimental purposes made it easy to discover the essential requirements of a Tunny-breaking machine.

³⁶³ “The question was how to build a better machine-a question for an engineer. In a stroke of genius, the electronics expert Thomas H. Flowers solved all these problems”.

³⁶⁴ “Newman’s factory of monstrous electronic computers, dedicated to breaking Tunny, was affording a glimpse of the future”.

popolari londinesi di Whitechapel e Poplar – il che lo rendeva un autentico *cockney*, come il suo accento testimoniava. “Fu sempre una specie di emarginato a Bletchley Park, dove NQOC stava per *Not Quite Our Class*” [Copeland, 2012, p. 129, trad. it.].

Studiò da apprendista al *Royal Arsenal* a Woolwich, prese una laurea in ingegneria all'università serale e ottenne un lavoro nel 1926 al *Telephone Branch* del *Post Office*. Flowers sapeva che il futuro era l'elettronica. Entrato alla fine degli anni Trenta nel *Research Branch* delle Poste, a Dollis Hill, e scalatone rapidamente i ranghi, “stava costruendo una memoria digitale elettronica gigantesca” [Copeland, 2012, p. 130, trad. it.] per sostituire l'antiquato e lento sistema a relè, soggetto a problemi di connessione, che regolava le installazioni telefoniche britanniche.

. Ricorda Flowers nel suo saggio retrospettivo *The Design of the Colossus* [Flowers, 1983]: “Penso che la mia reazione a questo lavoro sia stata quella di qualsiasi giovane ingegnere dell'epoca: *come possiamo fare un sistema migliore?*” [Flowers, 1983, p.241 trad. mia enfasi originale]³⁶⁵.

La risposta era nell'elettronica. Iniziatosi a interessare agli interruttori elettornici a metà degli anni '30, “Entro il 1939 sentivo di poter dimostrare ciò che fino ad allora era solo un sospetto: era possibile costruire un equivalente elettronico di qualsiasi macchina elettromeccanica di commutazione o elaborazione dei dati” [Flowers, 1983, p.241 trad. mia]³⁶⁶.

Allo scoppio della guerra, dopo aver lavorato ai sistemi radar e, nell'estate del 1940 durante la Battaglia d'Inghilterra, ai circuiti di comunicazione che trasmettevano al comando della RAF la posizione degli aerei della Luftwaffe, nel febbraio del 1941 Flowers fu contattato da Sir W. G. Radley, direttore del *Post Office Research Station*, per occuparsi di un problema per conto di Bletchley Park [Copeland, 2012, pp. 130-131, trad. it.; Randell, 1980, p. 56]. Qui collaborò con Turing sul progetto Enigma e, grazie alle pressioni di quest'ultimo, ottenne il titolo di MBE (*Member of the Order of the British Empire*) [Cfr. Copeland, 2012, pp. 130-132]. Quando Newman iniziò ad avere problemi con le sue macchine, “Turing disse ‘Flowers’” [Copeland, 2012, p. 132, trad. it.].

La macchina di Newman, come abbiamo visto era una realizzazione in hardware del metodo di Tutte.

³⁶⁵ “I think my reaction to this work was that of any young engineer of the time: how can we make a better system?”

³⁶⁶ “By 1939 I felt able to prove what up to then I could only suspect: that an electronic equivalent could be made of any electromechanical switching or data-processing machine”.

Le due ruote, i due *loop* di nastro da telescrivente, di Heath Robinson il *message-tape* e il *chi-tape*, processate insieme erano l'equivalente del messaggio da decifrare e dell'ipotesi statistico-matematica sulle impostazioni della macchina Tunny.

Ma Newman voleva risolvere i problemi della macchina, in primis quello della "temperamentalità" [*the machine was temperamental*] (del sistema a due ruote, cioè, non sempre a uno stesso input, Heath Robinson restituiva lo stesso output) [Edgerley, 2006, p. 275; Cfr. Reeds, 2015, p.535].

Una testimonianza dell'approccio innovativo basato sull'elettronica di Flowers si può trovare in due minute, datate 1 e 13 marzo 1943, scritte da Newman e indirizzate a Edward W. H. "Jumbo" Travis (1988-1956), responsabile delle operazioni di crittoanalisi. Nella prima, Newman riferisce che Flowers aveva presentato idee per una macchina dallo "schema molto più ambizioso" rispetto a quelli di Heath Robinson. Flowers aveva proposto l'idea di una macchina totalmente diversa, "in cui il messaggio e le ruote da confrontare con esso sarebbero impostati su valvole elettroniche, tramite l'uso di relè. Ciò comporterebbe l'impiego di circa 5000 valvole e 2500 relè. [...] Il movimento meccanico verrebbe completamente eliminato, e si potrebbero raggiungere velocità comprese tra 5000 e 10.000 operazioni al secondo", al posto dei 2000 caratteri al secondo raggiunti da Heath Robinson [Newman, minuta a J. Travis 1 marzo 1943, in Reeds, 2015, p.536 trad. mia]³⁶⁷.

Malgrado Newman percepisse la bontà di questo approccio, e ne consigliasse una sperimentazione parallela, esprimeva dubbi sull'effettiva affidabilità di un dispositivo così concepito, "il calore generato da tali valvole equivarrebbe a quello di 8 stufe elettriche a due elementi!". Sugeriva di dare priorità alle macchine "a nastro più semplici" (*simpler tape-machine*) [Newman, minuta a J. Travis 1 marzo 1943, in Reeds, 2015, p.536 trad. mia]³⁶⁸. A giocare a sfavore dell'idea di Flowers c'era l'impiego estensivo di valvole elettroniche, ritenute inaffidabili. La proposta dell'ingegnere del Post Office fu accolta con incredulità [Cfr.Reeds, 2015, p.535; Boden, 2006, p.159; Copeland, 2006d, p.74].

Nella seconda minuta di Newman, si legge che Flowers avrebbe proposto un secondo tipo di macchina, in cui "usare il nastro del messaggio solo per il messaggio e le valvole solo per

³⁶⁷ "Flowers, of the P.O., has produced a suggestion for an entirely different machine, in which the message, and the wheels to be compared with it, would be set up on valves, by means of relays. This would involve 5000 or so valves, and about 2500 relays. [...] Mechanical movement would be avoided altogether, and speeds of 5000 to 10,000 a second would be possible. This is, of course, a much more ambitious scheme".

³⁶⁸ "The heat the valves would generate would be equal to that of 8 two-bar electric fires!".

le ruote fisse”³⁶⁹ [Newman, minuta a J. Travis 13 marzo 1943, in Reeds, 2015, p.538 trad. mia]. Questa soluzione, prosegue Newman, avrebbe risolto una possibile obiezione alla prima proposta di macchina, quella legata al design logico-implementativo del dispositivo - che sarebbe risultato meno “flessibile” [*lack of flexibility*]³⁷⁰ - ma lasciava ancora irrisolte le perplessità legate all’uso estensivo di valvole elettroniche. “Sono ancora necessarie circa 1700 valvole [la seconda cifra è illeggibile] e riconoscono che ci saranno problemi iniziali di messa a punto.” [Newman, minuta a J. Travis 13 marzo 1943, in Reeds, 2015, p.538 trad. mia]³⁷¹.

La scelta progettuale per le macchine di decrittazione, sotto la supervisione di Wynn-Williams, era quella per cui si preferiva impiegare “il minor numero possibile di valvole, privilegiando invece i relè elettromagnetici” [Randell, 1980, p.64 trad. mia]³⁷². Flowers invece era fortemente convinto che la strada giusta da perseguire fosse quella dell’elettronica. “La cosa fondamentale di Flowers era che non gli importava di quante valvole usasse” [Chandler, Coombs, trascrizione intervista 10 novembre 1975, in Randell, 1980, p.64 trad.mia]³⁷³. Flowers basava la sua fiducia verso le valvole sulla sua esperienza nell’uso di queste nella telefonia. Come egli stesso ricorda:

La mia proposta, fatta nel febbraio del 1943, fu accolta inizialmente con notevole scetticismo. La prima reazione fu che una macchina con quel numero di valvole, chiaramente necessario, sarebbe stata trop po inaffidabile per essere utile. Fortunatamente, questa critica poteva essere superata grazie all’esperienza del Post Office, che utilizzava migliaia di valvole nella propria rete di comunicazione. Queste valvole non erano soggette a movimenti o manipolazioni, e l’alimentazione non veniva mai interrotta. In tali condizioni, i guasti delle valvole erano molto rari [Flowers, 1983, p.244 trad. mia]³⁷⁴.

³⁶⁹ “For the more ambitious machine they now propose to use tape for the message and valves only for the fixed wheels”.

³⁷⁰ L’obiezione di Newman alla prima proposta di Flowers - che prevedeva una macchina interamente elettronica, priva di nastri e basata solo sulla rappresentazione valvolare del messaggio cifrato e del chi-tape - riguardava la sua scarsa flessibilità nell’eseguire le cosiddette *compound runs*, ovvero confronti statistici nel *de-chi stream* tra più ruote simultaneamente. Tali operazioni, fondamentali per propagare l’impostazione di alcune ruote su altre, richiedevano l’elaborazione incrociata di molteplici flussi derivati dal messaggio cifrato. In assenza di nastri, una macchina con capacità limitata (stimata in circa 5.000 valvole, una per carattere) non avrebbe potuto gestire messaggi di lunghezza operativa normale, tra 1.500 e 6.200 caratteri, rendendo impraticabili le *compound runs* su traffico reale. Cfr. Reeds, 2015, pp.538-539.

³⁷¹ “About 1700 [second digit illegible] valves are still needed & they recognise that there will be teething troubles”.

³⁷² “This approach was very much in contrast to that of Wynn-Williams, who preferred to use as few valves as possible, favoring electromagnetic relays instead”.

³⁷³ “The basic thing about Flowers was that he didn't care about how many valves he used”

³⁷⁴ “My suggestion, made in February 1943, was received with considerable scepticism. The first reaction was that a machine with the number of tubes that was obviously going to be needed would be too unreliable to be useful. Fortunately, this criticism could be defeated by the experience of the Post Office using thousands of tubes in its communication network. These tubes were not subject to movement or handling, and the power was never switched off. Under these conditions tube failures were very rare”.

La priorità di Bletchely Park rimaneva all'inizio del '43 la costruzione delle Robinson, le macchine a doppio nastro. Per quanto riguarda la macchina elettronica a un nastro solo, "si lasciò la decisione a coloro che l'avrebbero costruita". [Flowers, 1983, 244 trad. mia]³⁷⁵. Non c'erano garanzie che questa, ispirata a una tecnologia mai impiegata nell'uso crittoanalitico, fosse terminata prima della fine della guerra. Qui, grazie all'incredibile abnegazione del team di tecnici e ingegneri di Flowers a Dollis Hill, si compì uno degli eventi all'apparenza improbabili, uno della serie di "serie di fortunate coincidenze" [Flowers, 1983, p.240 trad. mia]³⁷⁶ che hanno costellato la nascita di questa nuova macchina,

Nel frattempo, gli ingegneri di Dollis Hill, avendo deciso di realizzare un processore completamente elettronico, impiegarono solo 11 mesi per mettere in servizio la prima macchina - un'impresa resa possibile grazie all'assoluta priorità loro assegnata per ottenere materiali e servizi, e agli sforzi prodigiosi del personale di laboratorio, molti dei quali non facevano altro che lavorare, mangiare e dormire per settimane e mesi, con la sola eccezione di mezza giornata a settimana concessa per le necessità della vita come parlare con le mogli e tagliarsi i capelli [Flowers, 1983 trad. mia]³⁷⁷.

Il 18 gennaio 1944, il dispositivo - caricato dai ragazzi di Flowers sul retro di un furgone - fu trasferito da Dollis Hill a Bletchley Park, suscitando l'incredulità di chi vi posava lo sguardo [Copeland, 2012, pp. 75, 132]. Era una macchina enorme, grande quanto una stanza e pesante circa una tonnellata: un vero colosso. Proprio *Colossus* fu il nome con cui le operatrici Wren la ribattezzarono [Copeland, 2006a, p. 2; Flowers, 1983]. "Il gigantesco computer di

³⁷⁵ What to do about a one-tape machine with all-electronic processing would be left to those who would have to make it to decide.

³⁷⁶ "That Colossus occurred at all was the result of a series of lucky chances to which one more is added if because of it I am to be numbered among the computer pioneers". Copeland fa notare la serie, che sembra quasi incredibile, di coincidenze fortunate che hanno permesso a Colossus di venire alla luce - e di incidere, oltre che nella storia della computazione, anche sulle sorti della guerra al nazifascismo. "La stessa esistenza di Colossus fu talmente contraria ad ogni previsione da essere quasi un puro colpo di fortuna". Se Radley non avesse sostenuto Flowers nel suo uso delle valvole, si sarebbe proseguito con la tecnologia elettromeccanica; se Turing non avesse fatto il suo nome a Newman, Flowers sarebbe rimasto tranquillamente al Post Office; e se l'ingegnere britannico non fosse riuscito a sgattaiolare via - attraverso l'Olanda - dalla Germania in cui era stato incautamente mandato prima dello scoppio della guerra, Colossus non sarebbe mai nato [Cfr. Copeland, 2012, pp. 135-136, trad. it.]. Come riconosce, senza arroganza, retrospettivamente Flowers: "Quegli eventi fortunati cambiarono il corso della Seconda guerra mondiale. Se non si fossero verificati, ora la storia conoscerebbe una devastazione dell'Europa e un numero di morti molto più alto di quello che si ebbe" [Flowers, 2006, p. 82, in Copeland, 2012, p. 136, trad. it.].

³⁷⁷ "Meanwhile, the Dollis Hill engineers, having decided to make an all-electronic processor, took only 11 months to get the first machine into service, a feat made possible by the absolute priority they were given to command materials and services and the prodigious efforts of the laboratory staff, many of whom did nothing but work, eat, and sleep for weeks and months on end except for one half day per week that they allowed themselves for the necessities of life such as talking to their wives and getting their hair cut".

Flowers entrò in servizio attivo contro i messaggi tedeschi il 5 febbraio 1944” [Copeland, 2017, p. 155, trad. it.].

Il primo modello, il Mark I, conteneva 1600 valvole e operava a 5000 caratteri al secondo [Copeland, 2006d, p. 74; Flowers, 1983]. I modelli successivi, denominati Colossus II o Mark 2, arrivarono da Dollis Hill a Bletchley Park il 4 marzo 1944: impiegavano 2400 valvole, processavano cinque flussi DOT-CROSS in parallelo e raggiungevano la velocità di 25000 caratteri al secondo [Copeland, 2006, p. 74; Flowers, 1983].

. Colossus era *de facto* “in parte un analogo elettronico della macchina Lorenz Tunny” [Flowers, 2006, p. 96 trad. mia]³⁷⁸.

La differenza fondamentale tra Colossus e Robinson è che su Robinson tutti i pattern sono perforati su nastri, mentre su Colossus solo un pattern è su nastro, l'altro è rappresentato elettricamente [...] Colossus possiede un *bedstead* attorno al quale il nastro Z viene fatto scorrere tramite pulegge, in modo da essere scandito a 5000 lettere al secondo; e dei “trigger” nei quali possono essere impostati i pattern di chi, Mu e Psi [Good, Michie Timms, 1945, 13B, 13B (a), pp.25-26 trad. mia]³⁷⁹.

La realizzazione elettronica del *chi-stream* dava luogo a vantaggi indiscutibili rispetto alla tecnologia elettromeccanica alla base di Heath Robinson. Permetteva a Colossus di liberarsi dei vincoli imposti dalla sincronizzazione dei nastri, dell'usura, dell'inerzia e in definitiva della fisica newtoniana. “Colossus raddoppiò immediatamente la velocità dei decifratori” [Copeland, 2006d, p.75 trad. mia].

I vantaggi di questa macchina rispetto alla Robinson erano: (1) la sua velocità, un fattore di 25/2 quando erano disponibili 5 contatori per tutte le esecuzioni chi; (2) l'assenza di inerzia, che permetteva di interrompere un'esecuzione in qualsiasi momento e impostare le ruote sulle configurazioni desiderate; (3) l'elevata affidabilità derivante dall'uso esclusivo di valvole al posto dei relè e dall'eliminazione dei nastri sincronizzati. [Cfr. Good, Michie Timms, 1945, 13B, 13B (a), pp.25-26]

Al cuore di Colossus c'era l'idea che l'implementazione fisica in loop di un insieme di dati - nello specifico, il messaggio tradotto in un pattern di fori su nastro - non fosse indispensabile.

³⁷⁸ “Much of Colossus was an electronic analogue of the Lorenz Tunny machine”.

³⁷⁹ “The fundamental difference between Colossus and Robinson is that on Robinson all patterns are punched on tapes, whereas on Colossus only one pattern is on a tape, the other being represented electrically [...] Colossus has a “bedstead” round which the Z tape is driven by pulleys so as to be scanned at 5000 letters per second; and “triggers” in which chi, Mu, Psi patterns may be set up”.

Anzi, era proprio questo vincolo materiale, che poteva invece essere superato mediante l'impiego dell'elettronica per la rappresentazione dei dati corrispondenti, a causare i problemi che affliggevano Heath Robinson. Le parole di Flowers sono inequivocabili:

Non era il nastro in sé ad essere essenziale per l'elaborazione, ma i dati contenuti nel nastro. Tutto ciò che il "nastro chiave" faceva era conservare dati pre-generati, affinché potessero essere letti da un processore elettronico a una velocità molto maggiore. Un equivalente elettronico della macchina che aveva generato originariamente quei dati - e non avevo dubbi che si potesse costruire - avrebbe potuto lavorare abbastanza rapidamente da fornire direttamente i dati al processore, rendendo così superfluo il nastro chiave. Sarebbe stato necessario solo un unico anello, quello del nastro messaggio, che poteva essere trascinato da pulegge lisce ad alta velocità senza rischio di danneggiamento. [Flowers, 1983, p.244 trad. mia]³⁸⁰.

"L'idea di Flowers era quella di immagazzinare all'interno della macchina - nel linguaggio moderno si direbbe registrare in memoria - le chiavi del sistema «Pesce», e di farlo in forma elettronica" [Hodges, 1985, p. 348, trad. it.].

Al posto dell'impiego di tecnologia elettromeccanica, come i relè, veniva adottato l'uso di valvole termoioniche [*thermionic valves*] [fig. 16] per eseguire le operazioni computazionali e le logiche booleane necessarie alla decrittazione.

³⁸⁰ "It was not the paper tape that was essential to the processing, but the data on the tape. All that the "key" tape was doing was storing pregenerated data so that they could be read into an electronic processor at a much higher speed. An electronic equivalent of the machine that had originally generated the data-and I had no doubt that such an equivalent could be made-could work fast enough to issue the data directly to the processor and thus render the "key" tape unnecessary. Only one loop, that of the message tape, would be needed, and that could be driven by smooth-faced pulleys at high speed without fear of damage".



Fig. 16 Una valvola termoionica impiegata negli anni'40 [fonte Wikipedia]

Un relè è un interruttore elettromeccanico a forma di U, che utilizza una bobina avvolta parzialmente attorno al nucleo e un'ancora mobile in ferro che collega le due estremità aperte della U. L'ancora si muove tra due posizioni, ruotando su un perno. Quando non c'è corrente, una molla mantiene l'ancora in una posizione tale da lasciare un'apertura nel circuito: il circuito aperto, cioè la posizione OFF, rappresenta lo 0 digitale; quando invece la bobina è percorsa da corrente, si comporta come un magnete, attrae l'ancora e chiude il circuito, attivando lo stato ON, corrispondente a 1 digitale. In questo modo, il relè rappresenta un bit come presenza o assenza di connessione elettrica [Flowers, 2006, p. 92]. Come ha mostrato Shannon nella sua tesi magistrale, tali relè potevano essere interpretati come implementazioni fisiche di operazioni logiche binarie.

Tutti i calcolatori che utilizzavano dispositivi elettromeccanici potevano essere ripensati in forma elettronica. La via della computazione invisibile, basata sui flussi di elettroni e virtualmente libera dalle pastoie della meccanica classica, era ormai tracciata. Su di essa, come vedremo avanti negli sviluppi dal transistor al microprocessore, l'informazione avrebbe viaggiato a velocità fino ad allora impensabili.

“Una differenza fondamentale tra i due tipi di macchina è la velocità di funzionamento. La massima velocità a cui un interruttore a contatto metallico può operare è di circa mille

operazioni al secondo. Gli interruttori a valvole elettroniche possono raggiungere velocità di milioni di operazioni al secondo” [Flowers, 2006, p. 93, trad. mia]³⁸¹.

Le valvole termoioniche – note anche negli Stati Uniti come tubi a vuoto, *vacuum tubes* [Dyson, 2012, p. 76, trad. it.] – implementavano l’informazione controllando il flusso di elettroni all’interno di un tubo di vetro vuoto, tra catodo (l’elettrodo riscaldato che emette elettroni) e anodo (quello positivo che li raccoglie), tramite una griglia di controllo [*grid*]. Se la griglia era a potenziale positivo rispetto al catodo, il flusso passava: il circuito risultava acceso (stato ON), corrispondente a 1; se invece la griglia era negativa, il flusso veniva bloccato: il circuito era spento (OFF), corrispondente a 0. In questo modo, le valvole funzionavano come interruttori bistabili [Flowers, 2006 pp. 92–93].

Colossus, in particolare, utilizzava anche valvole *thyatron*, che potevano essere impiegate in alternativa al circuito trigger composto da due valvole a vuoto. Queste valvole – impiegate tutt’oggi – non sono a vuoto ma contengono una piccola quantità di gas, come il neon. “Viene commutato nello stato ON quando la griglia diventa positiva rispetto al catodo, ma può essere riportato nello stato OFF solo riducendo il potenziale dell’anodo per interrompere il flusso di corrente” [Flowers, 2006, p. 96, trad. mia]³⁸².

Le valvole thyatron, in Colossus, disposte su 12 anelli, simulavano elettronicamente i pattern della ruota χ [Flowers, 2006, p. 97; Wells, 2006, p. 124].

La memoria di Colossus comprendeva il nastro perforato a loop, che conteneva il messaggio cifrato da analizzare, e una configurazione di spinotti e prese che rappresentava i bit dei pattern delle ruote cifranti con cui il messaggio veniva confrontato. I dati semipermanenti e dinamici generati durante l’elaborazione venivano invece memorizzati elettronicamente tramite circuiti a valvole, tra cui i circuiti trigger Eccles–Jordan, inventati nel 1918 dai fisici britannici William Eccles (1875–1966) e F. W. Jordan (1882–1951) [Eccles, Jordan 1919], e le valvole thyatron stesse.

Colossus [fig. 17] non decifrava direttamente i messaggi cifrati, ma calcolava le impostazioni iniziali delle ruote della macchina Tunny, da cui dipendeva la generazione del flusso di chiave. Il messaggio cifrato, codificato in Baudot a cinque bit, veniva caricato su un nastro perforato disposto a loop continuo, come già avveniva in Heath Robinson, ma con

³⁸¹ “A fundamental difference between the two types of machine is the speed of operation. The fastest that metal-contact switches can be made to operate is about a thousand operations per second. Electronic valve switches can attain speeds of millions of operations per second”.

³⁸² “It is switched to the ON state when the grid becomes positive with respect to the cathode but can then be switched OFF only by reducing the anode potential to stop the flow of current”.

maggiore precisione e velocità. I fori di trascinamento [*sprocket holes*] generavano impulsi di temporizzazione [*timing pulses*] che sincronizzavano l'intera macchina. Ogni carattere veniva letto fotoelettricamente, convertito in impulsi elettrici e inviato a un codificatore elettronico.

Il codificatore era composto da un generatore di numeri – che riproduceva elettronicamente i pattern delle ruote Tunny (inizialmente le sole ruote χ , in seguito anche le ψ) – e da un sommatore, che combinava i numeri del messaggio e quelli delle ruote con un'addizione modulo-2 (operazione XOR). L'output risultante era composto da cinque flussi binari singoli, corrispondenti ai cinque impulsi del codice Baudot.

Tali flussi venivano inviati a unità logiche cablate, realizzate con configurazioni di spinotti e prese, che implementavano sommatore booleano, porte OR, AND, circuiti di ritardo e trigger. I segnali elaborati venivano confrontati secondo le relazioni previste dal metodo di Tutte (per esempio combinazioni tra due impulsi), al fine di identificare configurazioni delle ruote più coerenti con un messaggio linguistico sensato.

Solo le configurazioni che producevano un numero di coincidenze superiore a una soglia preimpostata venivano registrate: il contatore decimale a quattro cifre poteva essere impostato fino a 9999, e se il numero di coincidenze non raggiungeva tale soglia, la macchina non registrava nulla. Al contrario, se la soglia veniva superata, il conteggio e le posizioni delle ruote relative a quella configurazione venivano salvati tramite relè, e successivamente stampati da una macchina da scrivere elettromagnetica.

Una volta esaurite tutte le possibili combinazioni di impostazione iniziale delle ruote, si selezionavano quelle con i punteggi più alti, corrispondenti con alta probabilità alla configurazione realmente utilizzata dal mittente tedesco. I parametri così identificati venivano quindi utilizzati per la decifrazione completa del messaggio, effettuata su una macchina Tunny opportunamente modificata.

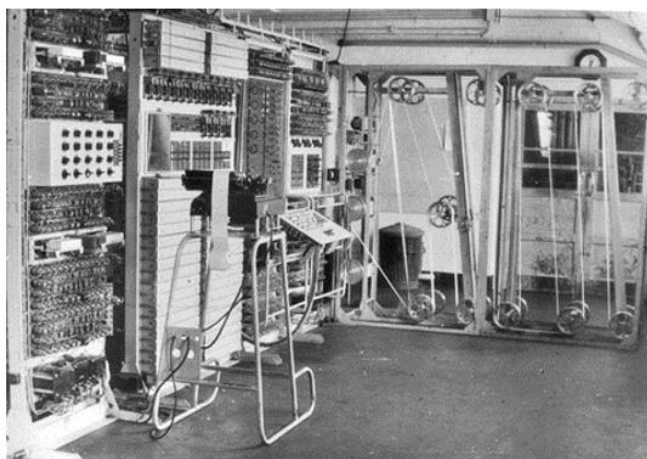
Le operazioni erano dirette da una unità di controllo principale [*master control*] - funzionalmente simile a una moderna CPU - che governava il ritmo dell'intero processo, sincronizzava la generazione dei pattern tramite loop di valvole thyatron, e avanzava le impostazioni iniziali delle ruote attraverso uniselettori elettromeccanici, attivati durante la porzione vuota del nastro. [Cfr. Flowers, 2006, pp.95-99].

Trovarsi davanti a una macchina del genere doveva suscitare uno stupore profondo e imprimere un'impressione destinata a durare, persino in un tempo segnato dalla concitazione e dall'urgenza della guerra. Lasciamo alle parole dei redattori del *General Report on Tunny* una descrizione vivida, che lambisce il lirismo, dell'effetto suscitato da Colossus: tanto più

significativa in quanto emerge in un documento pieno di specifiche tecniche, logiche e statistiche.

È un peccato che non sia possibile rendere adeguatamente l'idea del fascino di un Colossus in funzione: la sua mole imponente e la complessità apparente; la velocità vertiginosa del sottile nastro di carta che sfreccia tra le pulegge lucenti; il piacere infantile dato da pulsanti come "*not-not*", "*span*", "*print main heading*" e altri marchingegni; la magia dello svelamento meccanico delle lettere, una per una (una nuova arrivata pensò di essere vittima di uno scherzo); l'azione inquietante della macchina da scrivere che stampava i punteggi corretti senza e oltre l'intervento umano; lo scorrere del display; i momenti di attesa febbrile culminanti con la comparsa improvvisa del punteggio tanto atteso; e i ritmi strani che caratterizzavano ogni tipo di esecuzione: l'ingresso solenne, la corsa breve ed erratica, la regolarità del *wheel-breaking*, il rettangolo monotono interrotto dai balzi impetuosi del ritorno del carrello, il chiacchiericcio frenetico del *motor run*, perfino il delirio grottesco di sciami di punteggi fasulli. Forse solo qualche poeta delle decrittazioni Tunny potrebbe rendere giustizia a questo tema; ma, sebbene fossero comparsi un'ode al Colossus e alcuni frammenti, sembravano tutti composti in tempi di frustrazione e scoramento, e consistevano quasi interamente in imprecazioni o maledizioni [Good, Michie, Timms, 1945, 51(j), p.327 trad, mia].

Fig. 17 Colossus



Tutti i calcolatori che utilizzavano dispositivi elettromeccanici potevano essere ripensati in forma elettronica. La via della computazione invisibile, basata sui flussi di elettroni e

virtualmente libera dalle pastoie della meccanica classica, era ormai tracciata. Su di essa l'informazione avrebbe viaggiato a velocità fino ad allora impensabili.

I successi di Colossus spinsero il Gabinetto di guerra britannico, nel marzo del 1944, a ordinare a Flowers la costruzione di altre quattro macchine. La richiesta salì rapidamente a dodici unità operative da consegnare entro il 1° giugno. Scrive Flowers: “Ricordo bene lo sgomento che suscitai quando dovetti rispondere senza mezzi termini che era impossibile” [Flowers, 1983, p.246 trad. mia]³⁸³. La richiesta dei vertici del governo e dell'esercito era motivata non solo dalle prestazioni della macchina, “ma anche dal sempre più diffuso uso tedesco dei messaggi da telescrivente, e dalle procedure operative ‘migliorate’ adottate dai tedeschi, che comportavano un maggiore sforzo da parte nostra per ottenere gli stessi risultati” [Flowers, 1983, p.246 trad. mia]³⁸⁴.

Il laboratorio di Flowers a Dollis Hill aveva già pronte alcune componenti, sufficienti a permettere la consegna di almeno due nuove macchine entro la data richiesta. Per le successive, Flowers propose di rilevare e riconvertire una fabbrica delle Poste a Birmingham per la produzione dei componenti, che sarebbero stati poi assemblati a Dollis Hill e consegnati operativi a Bletchley Park. In seguito, la fase di montaggio e collaudo fu trasferita direttamente alla Newmanry, che nel frattempo aveva ampliato il proprio organico fino a contare 27 uomini e 272 Wren [Good, Michie, Timms, 1945, 15B (c), p. 35].

Flowers e il suo team pianificavano un ritmo di consegna di circa una macchina al mese. “Churchill, da parte sua, diede a Flowers la massima priorità per tutto ciò di cui aveva bisogno” [Copeland, 2006d, p. 77, trad. mia]³⁸⁵. Dopo un ulteriore periodo di lavoro estenuante e ininterrotto - i visitatori alla fabbrica restavano “sconvolti dalla scala dello sforzo” [*staggered at the scale of the effort*] - Colossus II fu spedito a Bletchley Park il 4 maggio 1944. Rispetto al prototipo, questa macchina integrava un numero maggiore di valvole (fino a 2.500), nuovi pannelli e cavi ricollegabili, e offriva una flessibilità logica superiore, progettata appositamente per implementare i nuovi metodi di breaking dei pattern delle ruote Tunny sviluppati da Michie e Good [Cfr. Copeland, 2006d, p. 76].

³⁸³ The instructions had come from the highest level, and I well remember the consternation when I had to say flatly that it was impossible.

³⁸⁴ This sudden demand had stemmed not only from the success of the existing Colossus machine, but also from increasingly widespread German use of teleprinted messages, and from “improved” German operating procedures, which meant that we had to work harder to achieve the same result.

³⁸⁵ Churchill for his part gave Flowers top priority for everything he needed.

Le macchine di Flowers erano *special purpose*, realizzate per mettere in pratica i processi di decifrazione escogitati a Bletchley. Colossus, in un tratto di sorprendente modernità “aveva poca, se non nessuna, capacità di calcolo numerico; era una macchina che elaborava testo” [Ceruzzi, 2012, p.36 trad. mia]³⁸⁶. In modo mediato, la macchina di Flowers elaborava linguaggio reso discreto, digitalizzato e pseudorandomizzato.

Sebbene Jack Good ritenesse che le sue capacità di elaborazione booleana rendessero Colossus più flessibile di quanto previsto dal progetto originale, tanto da definire la macchina “tendente a essere di uso generale” [Good, 2006, p. 220]³⁸⁷.

Un altro dei protagonisti della Newmanry, Geoffrey Timms, tentò, dopo la guerra di sviluppare un sistema di moltiplicazioni in base 10 mediante una riconfigurazione della macchina, “ma si trattava di uno sforzo così forzato e complicato per i circuiti del Colossus che, di fatto, la macchina non riusciva a completare l’operazione prima che scattasse il successivo impulso di clock” [Budiansky, 2006, p.62 trad. mia]³⁸⁸. Colossus non può essere allora considerata una macchina generale.

Abbiamo visto come la cablatura e i sistemi elettronici implementassero parte delle funzioni di memoria fissa e semipermanente. Ma quest’ultima, in realtà, dipendeva in ultima istanza dalla componente umana. Emerge qui il tratto dell’interazione con l’essere umano, fondamentale nello sviluppo della computazione.

Colossus era privo di memoria di programma [*program memory*] in senso moderno. In esso “quella memoria fondamentale rappresentata dal cablaggio fisso, ricollegabile o selezionabile tramite interruttori, che configurava il controllo principale e l’instradamento dell’unità logico-aritmetica (ALU) coincideva esattamente con l’*interfaccia* del programmatore” [Wells, 2006, p. 132, trad. mia enfasi mia]³⁸⁹

Era l’addetto alla crittoanalisi, o la Wren di turno alla macchina, a realizzare fisicamente il cablaggio corrispondente alla sequenza delle operazioni. “Il crittoanalista-programmatore

³⁸⁶ “Colossus had little, if any, numerical calculating ability; it was a machine that processed text”

³⁸⁷ “Colossus could be regarded as the first large-scale electronic computer, albeit for a specialised purpose, but leaning towards being general-purpose”.

³⁸⁸ . Geoffrey Timms, one of the Newmanry mathematicians, after the war worked out an elaborate system of plugging up the machine so that it could in theory perform base-10 multiplication-but it was such a stretch and so complicated a process for the Colossus’s circuits that the machine could not in fact complete the operation before the next clock pulse turned over

³⁸⁹ This is the program store, the all important memory represented by the fixed, pluggable or switchable wiring of the main control and ALU routing. In other words, the program memory was exactly the programmer’s interface.

scriveva il programma in notazione simbolica su un foglietto e lo passava a una Wren [...] perché lo eseguisse” [Flowers, 1983, p.250 trad. mia]³⁹⁰

Colossus possedeva alcuni elementi hardware architetturealmente connessi a un sistema operativo – come gli interruttori e il *master control* –, tuttavia, poiché le istruzioni erano limitate a schemi di cablaggi e interruttori, l’uso di una istruzione poteva escludere l’uso di un’altra a causa dell’occupazione preliminare di una determinata presa, non vi era un’elaborazione sequenziale e, soprattutto rispetto al PC moderno l’elaborazione non era sequenziale [Cfr. Wells, 2006, p.132]³⁹¹. Nella macchina di Flowers possiamo considerare presente solo una memoria fisica per “programmi potenziali” [Wells, 2006, p. 132, trad. mia]³⁹². Senza l’effettivo essere umano impegnato a cablare i pannelli, rimanevano soltanto prese vuote e interruttori inermi.

Si può così osservare come quelle funzioni, corrispondenti a specifiche facoltà del cervello umano, che Turing e von Neumann volevano incorporare nelle macchine, in Colossus - sebbene quest’ultimo ne fornisse l’infrastruttura *in potenza* - rimanessero ancora delegate all’essere umano.

Colossus forniva un supporto quantitativo alla decrittazione, offrendo agli analisti un ausilio in termini di velocità. Tuttavia, il suo impiego e le sue caratteristiche anticipavano una delle qualità fondamentali dei computer moderni: la profonda interazione con l’essere umano. Un’interazione che colloca Colossus a pieno titolo nella traiettoria che stiamo ricostruendo. L’integrazione strettissima tra uomo e macchina nell’elaborazione, gestione e impiego di grandi volumi di informazione si realizzava, infatti, secondo tempi e modalità che, senza il supporto dell’artefatto tecnico, non sarebbero stati altrimenti possibili.

L’interazione e la sinergia tra analisti, Wrens e macchine realizzavano quella che soltanto la natura non biologica di Colossus ha impedito all’analista di Bletchley, collaboratore di Turing e figura di primo piano della Newmanry, nonché co-autore del *General Report on Tunny*, Jack Good, di definire una *simbiosi*.

Uno dei principali usi del COLOSSUS implicava una sinergia tra uomo e macchina che è stata raggiunta con i computer ordinari solo negli ultimi anni. (*‘Simbiosi’* è un termine improprio, poiché

³⁹⁰ “More often the cryptanalyst-programmer in charge of the Colossus program switches. machine wrote the program in symbolic notation on a slip of paper and gave it to a Wren to execute”.

³⁹¹ Per un interessantissimo confronto tra l’architettura, le caratteristiche e le funzioni di Colossus e quella di un moderno PC si veda

³⁹² “Colossus too had actual storage for potential programs”.

COLOSSUS non era un organismo biologico.) L'analista si sedeva davanti all'output della macchina per scrivere e dettava istruzioni a una Wren affinché apportasse modifiche ai programmi [Good, 1980, p. 41, trad. mia]³⁹³.

Le macchine restanti furono realizzate e consegnate come da programma da Flowers e dal suo team, "cosicché quando l'armistizio tedesco, undici mesi dopo il 1° giugno 1944, pose fine all'intera impresa, 10 macchine erano operative, e l'undicesima era in fase di collaudo" [Flowers, 1983]. La resa tedesca segnò la fine della necessità di nuove macchine, e fu lo stesso Churchill a ordinare la distruzione dei vari Colossus [Copeland et al., 2006, p. 172; Hilton, 2006, p.202]. Nell'aprile del 1946 il Government Code and Cypher School cambiò nome in *Government Communications Headquarters* (GCHQ) e si trasferì da Bletchley Park a Eastcote, nei sobborghi di Londra; un ulteriore trasferimento, nel 1952, portò uomini e apparecchiature a Cheltenham, fuori dall'area della capitale inglese [Cfr. Good, 2006, p. 173]³⁹⁴.

Sull'Europa calava una cortina di ferro e sul gigante metallico Colossus una di silenzio e segretezza. All'inizio degli anni '70, lo storico e *computer scientist* dell'Università di Newcastle, Brian Randell, nell'ambito di una sua indagine sul ruolo di Alan Turing all'interno di una più ampia storia delle origini della computazione digitale, dai lavori di Babbage in poi, si imbatté nell'esistenza di Colossus. Nel 1972 scrisse al Primo Ministro Edward Heath chiedendone la declassificazione, che fu rifiutata, ma accompagnata dalla promessa di una futura storia ufficiale. Colossus fu parzialmente desecretato solo il 20 ottobre 1975, quando al *Public Record Office* di Londra furono rese disponibili alcune fotografie e una didascalia esplicativa [Randell, 1980, pp. 48–49; Randell, 2017, pp. 67–72]. Il *General Report on Tunny* fu desecretato nel 2000, mentre un altro documento – *A Technical Description of COLOSSUS I*, redatto nell'agosto del 1973 da D. C. Horwood [Horwood, 1973] membro del Post Office che collaborò alla costruzione della macchina – fu desecretato solo nel 2003.

È comprensibile l'amarezza percepibile nelle parole retrospettive di Flowers, la stessa amarezza che spinse Turing a gettare la sua medaglia dell'Ordine Britannico in una cassetta

³⁹³ "One of the main uses of a COLOSSUS involved a synergy between man and machine that has been achieved with ordinary computers only during the last few years. ('Symbiosis' is a wrong term since COLOSSUS was not biological.) The analyst would sit at the typewriter output and call out instructions to a Wren to make changes in the programs".

³⁹⁴ Jack Good afferma che dopo l'ordine di distruzione impartito dai vertici politici e militari due Colossi sopravvissero, trasportati nelle nuove sedi della crittoanalisi britannica insieme a due copie di Tunny. Uno, noto come *Colossus Blue*, fu smantellato nel 1959 dopo 14 anni di servizio; l'altro rimase operativo fino al 1960. Negli ultimi anni furono impiegati per scopi di addestramento, mentre l'altro precedente rimane secretato. [Cfr. Good, 2006, p.173]

degli attrezzi [Hodges, 1985, p. 439 trad. it.] e addirittura Max Newman a rifiutare l'onorificenza [Newman, 2006, pp.195-196].

Quando, dopo la fine della guerra, mi fu detto che il segreto di Colossus sarebbe stato mantenuto indefinitamente, rimasi naturalmente deluso. Non avevo dubbi, una volta che il successo della macchina fu dimostrato, che Colossus rappresentasse una svolta storica, e che la sua pubblicazione avrebbe consacrato il mio nome nei circoli scientifici e ingegneristici - una convinzione confermata dall'accoglienza riservata all'ENIAC, l'equivalente statunitense reso pubblico subito dopo la fine della guerra. Dovetti sopportare tutti gli elogi rivolti a quell'impresa senza poter rivelare che l'avevo anticipata [Flowers, 2006, p.82 trad. mia]³⁹⁵.

L'ironia della storia ha voluto che quell'invisibilità che i protagonisti dell'ideazione e dell'implementazione di Colossus stavano conferendo alle forme della computazione digitale fosse la medesima a cui furono condannati loro stessi e la loro invenzione dalla coltre di segretezza imposta dal governo britannico.

Eppure, la sua influenza nella storia che andiamo delineando permane. Sebbene, come attesta il *General Report on Tunny*, "Colossus fu interamente un'idea del Sig. Flowers di Dollis Hill" [Good, Michie, Timms, 1945, 15B (b), p.35 trad. mia]³⁹⁶ - e il contributo di Turing fu tuttalpiù indiretto, ravvisabile nell'uso del *delta* comune al metodo non-*depth* di Tutte, che il macchinario implementava - "Flowers influenzò Turing profondamente" [Copeland, 2012, p. 158, trad. it.]. L'uso del digitale e dell'elettronica divenne così una possibilità concreta per dare realtà alla macchina che il matematico inglese aveva concepito nel 1936 per risolvere l'*Entscheidungsproblem*.

Il titolo, strappato a Colossus dal segreto imposto dalle autorità britanniche sulle attività di Bletchley Park, di primo calcolatore *digitale programmabile*, fu per lungo tempo assegnato a un'altra macchina, figlia delle esigenze della computazione militare divenuta di una complessità tale da risultare fuori dalla portata, in termini di mole e velocità di esecuzione, delle capacità umane: l'*Electronic Numerical Integrator and Computer*, o, con un acronimo - inaugurando quello stile di nomi per calcolatori che sarà reso famoso dai racconti di Isaac Asimov - ENIAC.

³⁹⁵ When, after the war ended, I was told that the secret of Colossus was to be kept indefinitely, I was naturally disappointed. I was in no doubt, once it was a proven success, that Colossus was a historic breakthrough, and that publication would have made my name in scientific and engineering circles - a conviction confirmed by the reception accorded to ENIAC, the US equivalent made public just after the war ended. I had to endure all the acclaim given to that enterprise without being able to disclose that I had anticipated it.

³⁹⁶ Colossus was entirely the idea of Mr Flowers of Dollis Hill.

Il *Ballistic Research Laboratory* aveva già iniziato, come abbiamo visto, a utilizzare strumenti meccanici per il calcolo delle traiettorie dei proiettili, al fine di produrre le tabelle di tiro. L'Analizzatore Differenziale di Bush aveva rappresentato a lungo il vertice della tecnologia in uso presso l'Aberdeen Proving Ground, ma la mole crescente di dati introdotta dalle innovazioni tecnologiche spinse il laboratorio balistico a rivolgersi anche alla *Moore School* dell'Università di Philadelphia.

John Mauchly (1907-1980), PhD in fisica, aveva trovato un posto da insegnante in una piccola università vicino Philadelphia, l'Ursinus College, e annoverava tra i suoi interessi il rapporto tra la climatologia e le macchie solari. Nel 1941 iniziò a seguire un corso di elettronica militare presso la Moore School. Gli venne proposto di diventare docente e accettò [Cfr. Dyson, 2012, p.83 trad. it.].

Fu in questo contesto che Mauchly entrò in contatto con le idee di Atanasoff - sebbene egli abbia sempre minimizzato tale influenza, anche a causa di una disputa legale sui brevetti [cfr. Dyson, 2012, p. 84] - il quale, come abbiamo visto, aveva iniziato sin dal 1935 a sperimentare con la computazione elettronica e che, nel 1941, aveva costruito una versione digitale ed elettronica dell'Analizzatore Differenziale, "che veniva usato occasionalmente, ma spesso rimaneva inattivo" [Burks, 1980, p. 314, trad. mia]³⁹⁷.

Mauchly aveva iniziato a far circolare internamente alla Moore School alcune idee sull'uso dell'elettronica in un memorandum nell'agosto del 1942, *The Use of High Speed Vacuum Tube Devices for Calculating* [Mauchly, 1942, in Randell, 1975, pp.329-332].

Tra le persone maggiormente colpite dalle considerazioni di Mauchly c'era un suo giovane studente laureato, l'ingegnere John Presper Eckert (1919-1995) "che era senza dubbio il miglior ingegnere elettronico della Moore School" [Goldstine, 1972, p.149 trad. mia]³⁹⁸.

È da rimarcare come Mauchly in questo documento, in cui elenca i vantaggi dell'uso dell'elettronica rispetto al funzionamento dell'analizzatore differenziale meccanico, utilizzi il termine *computer* [Cfr. Mauchly, 1942 in Randell, 1975, p.329]. Questa scelta sta quasi a sottolineare la differenza tra il macchinario che aveva in mente e l'utilizzo comune di *computer*, che come abbiamo visto, si riferiva agli esseri umani in carne ed ossa impegnati in elaborati calcoli con carta e matita.

³⁹⁷ "The Moore School had a differential analyzer, which was used some but often stood idle".

³⁹⁸ "J. Presper Eckert, Jr., who was undoubtedly the best electronic engineer in the Moore School". Un approfondimento biografico su Mauchly e Presper Eckert si trova in McCartney, 1999, pp.28-46.

La Moore School, intanto, era diventata, con lo scoppio della guerra, un centro di calcolo delle tabelle di tiro [Burks, 1980, p.314]. Qui vi fu assegnato con il grado di Luogotenente responsabile, Herman Goldstine. Questi, che aveva conosciuto Mauchly nell'autunno del 1941, ricorda:

Mauchly e io avevamo conversazioni abbastanza frequenti e reciprocamente stimolanti su questioni computazionali durante l'autunno del 1942. Questi colloqui mi servirono a comprendere meglio l'argomentazione di Mauchly riguardo al "grande incremento nella velocità del calcolo... se i dispositivi impiegati utilizzano mezzi elettronici per l'esecuzione del calcolo, poiché la velocità di tali dispositivi può essere resa molto superiore a quella di qualsiasi dispositivo meccanico [Goldstine, 1972, p.149 trad. mia]³⁹⁹.

Goldstine parlò del suo interesse per le idee di Mauchly a John G. Brainerd (1904-1988), il direttore della Moore School. Dopo un confronto con P.N. Gillon (1896-1966) *dell'Ordnance Department* del governo USA, i due convenirono sul fatto che il ministero di cui Gillon, che era particolarmente entusiasta del progetto, era rappresentante finanziasse un programma di sviluppo alla Moore School per la realizzazione di un calcolatore elettronico digitale per il BRL.

Il 2 aprile Brainerd preparò un rapporto da sottomettere a Gillon che diede forma alla base rapporto contrattuale tra l'Ordnance Department e la Moore School [Cfr. Goldstine, 1972, p. 149]. Il rapporto era intitolato *Report on an Electronic Diff. Analyzer* [Brainerd et al., 1943].

Il volutamente ambiguo "Diff." "esprimeva la transizione dal *differenziale* analogico alla *differenza* digitale, con implicazioni che da allora continuarono a far sentire il loro peso" [Dyson, 2012, pp.83-84 trad.it. enfasi originale]. La differenza a cui allude il titolo sembra quasi prefigurare il computer come elaboratore dell'informazione intesa, secondo la celebre definizione datane da Gregory Bateson, come "una *differenza che produce una differenza*" [Bateson, 1970, pp. 470 trad. it. enfasi originale], su cui è utile soffermarsi brevemente. Il concetto di informazione – insieme a quello di *dato* - analizzato con Shannon e Wiener e visto nella sua codifica digitale acquisirà qui via via maggior profondità di analisi nei modi in cui è stato implementato nei calcolatori.

³⁹⁹ "Mauchly and I had fairly frequent and mutually interesting conversations about computational matters during the fall of 1942. These talks served to emphasize to me Mauchly's point about the "great gain in the speed of the calculation... if the devices which are used employ electronic means for the performance of the calculation, because the speed of such devices can be made very much higher than that of any mechanical device."".

In che senso l'informazione, quell'informazione che abbiamo definito quantitativamente con Shannon e Wiener in relazione all'entropia e all'incertezza - che i calcolatori digitali permettono di creare, raccogliere, immagazzinare, elaborare, distribuire/trasmettere, utilizzare/consumare, cancellare/riciclare [Cfr. Floridi, 2014, pp.5-7 trad. it.] - è una differenza? Rispondere a questa domanda chiama in causa il concetto di *dato*.

Un dato è: “una mancanza di uniformità [...] in base all'*interpretazione diaforica* (*diaphora* è la parola greca per “differenza”) la definizione generale di dato è la seguente: Dato = def. l'essere *x* distinto da *y*, dove *x* e *y* sono due variabili non interpretate e il dominio resta aperto a ulteriori interpretazioni” [Floridi, 2011, p.85 trad. it. in Floridi, 2024 enfasi originale].

Questa interpretazione può darsi a tre livelli: 1) “Dati come *diaphora de re*, cioè come mancanza di uniformità nel mondo reale”, possibile riferirsi a loro come *dedomena* (il termine greco per “dato”). I *dedomena* sono “fratture nella trama dell'Essere”, proto-epistemici, anteriori a ogni rappresentazione e interpretazione. Non vanno confusi con i dati ambientali e non sono materiali. Funzionano come ipotesi o requisiti ontologici, come il *noumeno* di Kant o la *sostanza* di Locke. La loro presenza è dedotta empiricamente, ma non è direttamente esperibile. Questa interpretazione richiama il problema della possibilità di rappresentazione dei dati e della loro necessità di implementazione materiale. 2) “Dati come *diaphora de signo*”, come mancanza, o mancata percezione, della differenza tra almeno due *segnali*, come le differenze di carica, le variazioni elettriche o segni nel codice Morse. 3) “Dati come *diaphora de dicto*, cioè come mancanza di uniformità tra due simboli come ad esempio tra due lettere dell'alfabeto [Floridi, 2011, p.86 trad. it. in Floridi, 2024 enfasi originale].

L'informazione dipende dai dati – torneremo più avanti sull'ontologia di questi ultimi, fondamentale per comprendere cosa siano i *Big Data* – e i dati dipendono dalle differenze, intese come mancanza di uniformità. Si immagini un foglio su cui vengano progressivamente rimossi i segni scritti sopra fino a che non rimanga altro che il bianco della carta. Senza differenze, l'equazione di Shannon si applicherebbe al nulla: non vi sarebbe incertezza, non vi sarebbe entropia, e senza queste non vi sarebbe informazione. Parimenti, in assenza di tali differenze, le strutture di retroazione della cibernetica wieneriana non avrebbero alcun feedback da trasmettere.

“La dipendenza dell'informazione dalla presenza di dati ben formati sintatticamente, e quella dei dati dalla presenza di differenze variamente implementabili dal punto di vista fisico, spiegano perché l'informazione può essere così facilmente disgiunta dal suo supporto” [Floridi,

2011, p. 86, trad. it. in Floridi, 2024]. Si comprende allora l'importanza dell'uso dell'elettronica. Le valvole potevano far viaggiare quell'informazione da inserire nelle tabelle di tiro, che con fatica i calcolatori umani di Aberdeen, con carta e matita, cercavano di ricavare dalla massa di dati da considerare in relazione alla balistica, e farla viaggiare a una velocità non più soggetta ai vincoli della fisica classica. E questo, nella prospettiva che stiamo considerando, in un'ottica in cui, come elaboratore di informazione, non vi è più differenza formale o funzionale tra il neurone del cervello umano e il supporto artificiale dell'informazione.

Non deve stupire, allora, che una volta reso pubblico il 14 febbraio 1946, in una conferenza stampa destinata al grande pubblico e accolta con enorme clamore mediatico - basti ricordare l'articolo del *New York Times* dal titolo *Electronic Computer Flashes Answers, May Speed Engineering*, scritto da T. R. Kennedy Jr. e pubblicato il 15 febbraio 1946 [*New York Times*, 15 febbraio 1946] - con annessa dimostrazione in cui vennero presentate esibizioni spettacolari, come l'addizione di 5.000 numeri in un secondo e il calcolo di una traiettoria in soli 20 secondi, la stampa e i media dell'epoca abbiano descritto l'ENIAC con metafore quali *electronic brain*, *magic brain*, *wonder brain* [Martin, 1995].

Tornando alle vicende che hanno permesso la nascita dell'ENIAC, il documento del 2 aprile fu presentato da Brainerd, Mauchly e Presper Eckert il 9 aprile 1943 in una conferenza a cui erano presenti anche Oswald Veblen e il colonnello - e direttore del BRL - Leslie E. Simon (1900-1983). Nel meeting appena precedente a questo, Veblen, riferisce Goldstine, "dopo aver ascoltato per un breve momento la mia presentazione e mentre si dondolava sulle gambe posteriori della sedia, la riportò bruscamente a terra con un tonfo, si alzò e disse: "Simon, dai a Goldstine i soldi""[Goldstine, 1972, p.149 trad. mia]⁴⁰⁰.

Con questa dimostrazione di pragmatismo tipicamente americano, lontano dall'elefantiasi burocratica che attanagliava lo sviluppo dei calcolatori dall'altro lato dell'Atlantico, la strada verso la realizzazione del progetto si semplificò, malgrado le perplessità, analoghe a quelle che dovette subire Flowers su Colossus, espresse principalmente da T.H. Johnson (1900-1998) - capo del dipartimento dell'elettronica alla Moore School - intorno all'utilizzo di diverse migliaia valvole per la nuova macchina [Cfr. Goldstine, 1972, p.150; Eckert, 1980, p.526-527].

⁴⁰⁰ "His meeting had been preceded by others. One of these involved Simon, T. H. Johnson, Veblen, and myself at which Veblen after listening for a short while to my presentation and teetering on the back legs of his chair brought the chair down with a crash, arose, and said, "Simon, give Goldstine the money."".

Il 31 maggio i lavori erano pronti a partire. Il 5 giugno del 1943 il contratto che sanciva l'inizio dei lavori divenne in essere. Lo stanziamento iniziale fu di 61.700 dollari per sei mesi - al completamento della macchina il 30 settembre 1945, il costo era levitato a 487,125 dollari [Haigh et. al, 2016]. "Il progetto PX [il nome in codice assegnato] era nato" [McCartney, 1999, p. 61, trad. mia]⁴⁰¹

In una riunione immediatamente precedente, in cui erano presenti tra gli altri Goldstine, Johnson per il BRL oltre a Brainerd, Mauchly e Presper Eckert per la Moore School, Gillon che la presiedeva assegnò alla macchina il nome di *Electronic Numerical Integrator And Computer*, ENIAC [Goldstine, 1972, p. 150].

Il *computer* di Mauchly era diventato un *computer*. Ricorda Presper Eckert come "il nome ENIAC dove la a "I" sta per "Integrator" e fu scelta per convincere il Pentagono che ciò che il Ballistic Research Laboratory stava ottenendo avrebbe calcolato tavole di tiro, che nel 1943 rappresentavano il bisogno più urgente dell'Ordnance" [Presper Eckert, 1980, p.526 trad. mia]⁴⁰².

L'ENIAC era una macchina gigantesca, occupava una stanza di 16 metri per 10, era pesante svariate tonnellate, ed era composto da 40 pannelli disposti a U in modo tale da permettere agli operatori di potersi muovere all'interno. [Goldstine, Goldstine, 1946]. Come ricorda Harry Reed, un matematico del centro di Aberdeen, ENIAC era a suo modo un Personal Computer: "Oggi pensiamo al personal computer come qualcosa che puoi portarti in giro. L'ENIAC era in pratica un computer dove potevi vivere" [Dyson, 2012, p.85 trad. it.]. I pannelli contenevano 18000 (per la precisione 17468) valvole e 1500 relè. Questi erano divisi in 30 unità, ognuna per una funzione particolare che il calcolatore doveva assolvere [Goldstine, Goldstine, 1946].

Le altre parti di ENIAC – come annotò Arthur Burks (1915-2008), un filosofo e logico che si occupò di calcolatori durante la guerra collaborando anche con von Neumann- nel suo diario l'aprile del '44 – erano "20 accumulatori, tre tavole di funzioni e uno ciascuno dei seguenti componenti specializzati: un moltiplicatore, un'unità per la divisione e l'estrazione della radice quadrata, un trasmettitore di costanti, un'unità di ciclo, una stampante e un *master programmer*" [Haigh et al., 2016, p.45 trad.mia]⁴⁰³.

⁴⁰¹ "That was it. Project PX was born".

⁴⁰² "The name "ENIAC," where the "I" stands for "integrator," was devised to help sell the Pentagon that what the BRL was getting would compute firing tables, which were, in 1943, the greatest need of Ordnance".

⁴⁰³ "Burks' notebook reports that during that meeting "plan of # of units, etc., of ENIAC [were] finally arrived at." ENIAC was to consist of 29 units spread across 36 panels: 20 accumulators, three function tables, and one each of

A lavori terminati si aggiunse una unità di avvio oltre alla struttura a 40 pannelli per le 30 unità e si adottò una architettura logica e implementativa parallela e modulare.

Man mano che i lavori progredivano, il ruolo centrale di quell'unità venne sostituito da un sistema distribuito, in cui le informazioni di programmazione erano diffuse tra le varie unità di ENIAC. Nella sua configurazione finale, ENIAC consisteva in larga parte in un plotone di accumulatori che marciavano affiancati al ritmo dettato dall'unità di ciclo, ricevendo di tanto in tanto nuovi ordini da un'unità chiamata "master programmer" [Haigh et al., 2016 p.36, trad.mia]⁴⁰⁴.

Le operazioni aritmetiche erano incorporate direttamente nella macchina: ogni accumulatore utilizzava contatori circolari basati su 10 flip-flop bistabili realizzati con valvole termoioniche, in grado di rappresentare numeri decimali a 10 cifre più il segno. Addizione e sottrazione venivano eseguite tramite impulsi paralleli che codificavano le cifre, mentre la sottrazione sfruttava un sistema di numeri complementari per facilitare i calcoli.

Delle tavole di funzioni utilizzavano matrici di resistenza per memorizzare fino a 104 valori tabellari di 12 cifre e due segni ciascuno, permettendo un rapido richiamo di dati preimpostati durante le operazioni. Queste erano state progettate da Jan Rajchman – della RCA (*Radio Corporation of America*), ulteriore segno di quella convergenza tecnologica che ha caratterizzato e caratterizza lo sviluppo dei calcolatori. La RCA era stata inizialmente contattata per la fornitura delle valvole termoioniche destinate all'ENIAC, ma Vladimir K. Zworykin (1889–1982), allora direttore dei laboratori di ricerca RCA a Princeton, non volle impegnarsi in un progetto così complesso e considerato poco affidabile, sebbene la compagnia offrì consulenza ed expertise ai progetti di computazione bellica [Dyson, 2012, p.85 trad. it.].

Emigrato dalla Russia nel 1919, Zworykin fu un pioniere delle tecnologie radiotelevisive e delle tecnologie elettroniche per la conversione tra fotoni ed elettroni che rese possibile la televisione commerciale. Distinse tre fasi nello sviluppo dell'elettronica: nella prima – con l'invenzione dell'*audion* di Lee de Forest nel 1906, la corrente elettronica nei tubi a vuoto era controllata come il vapore da una valvola meccanica; nella seconda (anni '20), si sfruttava la direzionalità dei fasci elettronici nei tubi a raggi catodici; nella terza (anni '30), si otteneva una

the specialized multiplier, a divider-square rooter, a constant transmitter, a cycling unit, a printer, and a master programmer".

⁴⁰⁴ "As work progressed, that unit's central role was replaced by a distributed system in which programming information was spread around ENIAC's various units. As ENIAC was eventually built, it consisted in large part of a platoon of accumulators marching side by side to the beat of the cycling unit and occasionally receiving new orders from a "master programmer"".

strutturazione temporale o spaziale del fascio, come nei magnetroni o nei tubi d'immagine. [Cfr. Zvorykin, 1951; Dyson, 2012, pp. 76-79 trad. it.].

La quarta fase doveva essere l'evoluzione dell'elettronica impiegante le valvole termoioniche. Durante la guerra, Jan Rajchman, sviluppò il *Computron*, un tubo aritmetico digitale a 64 pin e 737 componenti, ma impossibile da correggere una volta sigillato, e il *Selectron*, una memoria digitale a accesso casuale da 4096 bit, funzionante con tecnologia a tubo a vuoto. Una sorta di antenato *termoionico* del transistor e del *microchip*. Fu proprio il potenziale del Selectron che convincherà von Neumann che la strada per il calcolo digitale passava attraverso l'elettronica. [Cfr. Rajchman, 1980, pp. 465-469; Dyson, 2012, pp.79-82 trad. it.].

Tornando alle caratteristiche di ENIAC, in esso venti processori comunicanti [*accumulators*] erano assistiti da un'unità di calcolo per le operazioni di moltiplicazione e divisione e radice quadrata, sfruttando tabelle elettroniche di prodotti parziali e procedure di sottrazioni successive, rispettivamente [Cfr. Dyson, 2012, p.84 trad. it.].

La comunicazione tra le unità avveniva tramite reti di cavi chiamate *digit trunks*, per i dati numerici, e *program trunks*, per i segnali di controllo e sincronizzazione; ogni trunk era composto da 11 conduttori, consentendo la trasmissione parallela simultanea di cifre e comandi, garantendo coordinazione e alta velocità operativa [Goldstine, Goldstine, 1946]

La programmazione era distribuita localmente tramite una serie di *program controls*, cioè set di interruttori e flip-flop associati a ciascuna unità, che determinavano quali operazioni eseguire e gestivano l'inizio e la fine di ogni calcolo. La sequenza temporale delle istruzioni, pur nell'implementazione parallela, era coordinata dal *master programmer*, un'unità centrale costituita da contatori stepper e decade che permettevano di contare impulsi e modificare le connessioni di programma, abilitando cicli, iterazioni e controlli condizionali, come la scelta del flusso in base al valore o segno di cifre specifiche. L'input e l'output venivano gestiti con dispositivi IBM a schede perforate, integrati nel sistema grazie a un'accurata sincronizzazione con l'unità di cycling che emetteva impulsi di clock a circa 5 kHz, suddivisi in cicli di 1/5000 di secondo, garantendo l'ordinata esecuzione delle operazioni. [Cfr. Goldstine, Goldstine, 1946; Dyson, 2012, p.84 trad. it.]

Chi operava su ENIAC comunicava con la macchina attraverso quelle schede perforate che, come abbiamo visto, circa vent'anni dopo erano diventate il simbolo che concretizzava i timori di una visione di società ridotta a sistema automatizzato e di essere umano scarnificato

sino ad essere un mero aggregato di informazioni. Qui è possibile individuare un'ulteriore traccia della confluenza di tecnologie che concorrono a dar forma ai calcolatori.

Le schede perforate rappresentano un metodo di implementazione materiale dell'informazione simbolica che è durato fino agli anni '70 del Novecento [Chalmers, 2022]. "L'idea di utilizzare un meccanismo di controllo a sequenza affinché una macchina possa eseguire azioni complesse, costituite da una serie di operazioni di base magari piuttosto semplici, è un concetto molto antico" [Randell, 1975, p.4 trad. mia]⁴⁰⁵. Il primo di tali meccanismi fu il cilindro rotante con perni, ancora oggi visibile nelle scatole musicali. L'origine di tali meccanismi può essere fatta risalire al XIV secolo, ai cilindri musicali impiegati per controllare i movimenti di figure in miniatura che decoravano grandi orologi da chiesa. Successivamente furono impiegati per controllare organi a canne e, ad esempio, alla fine del XVI secolo Salomon de Caus (1576–1626) utilizzò un meccanismo simile per regolare l'esecuzione simultanea di suoni e movimenti meccanici. Tuttavia, tali dispositivi erano integrati alla macchina che controllavano. Non costituivano ancora una forma autonoma di programma.

Un salto concettuale avvenne nel XVIII secolo con l'invenzione di Basile Bouchon (attivo intorno al 1725), che impiegò un nastro perforato per automatizzare la selezione dei fili per i telai da seta. Questo principio fu perfezionato da Jean-Baptiste Falcon (morto nel 1765). Questi sostituì il nastro con una serie di schede perforate collegate, aumentando la capacità di controllo e la complessità dei motivi tessili realizzabili. Verso il 1750, Jacques de Vaucanson (1709–1782), celebre per i suoi automi, progettò il primo telaio completamente automatico, tornando però a un sistema meno flessibile con un cilindro perforato al posto delle schede.

Sarà Joseph-Marie Jacquard (1752–1834), coadiuvato da Jean Antoine Breton (17775-1836), a realizzare nei primi anni del XIX secolo il primo telaio automatico realmente efficiente, fondato su schede perforate modulari e intercambiabili. Il telaio Jacquard non solo rivoluzionò l'industria tessile, ma fornì anche il modello di controllo programmato che ispirò Charles Babbage (1791–1871) nel concepimento della sua *Analytical Engine*.

A cavallo tra la fine dell'800 e l'inizio del '900 l'aumento demografico aveva reso obsolete le tecniche tradizionali, manuali, di analisi, gestione e tabulazione dei dati riguardanti la popolazione necessarie al censimento nazionale decennale. Dopo alcuni esperimenti, come la macchina di Seaton che impiegava un sistema di rulli per aiutare gli impiegati nell'allineamento dei moduli, si rese necessario un sistema più efficiente. Herman Hollerith (1860-1929), che era

⁴⁰⁵ "The idea of using a sequence control mechanism so that a machine can perform very, complex actions consisting of a sequence of perhaps quite simple basic operations is a very old one indeed".

stato collaboratore di John Shaw Billings (1838-1913) - dirigente per la pianificazione statistica nel censimento del 1880 - ebbe l'idea, ispirata forse da Jacquard, forse dallo stesso Billings, di impiegare le schede perforate per meccanizzare la tabulazione dei dati.

Hollerith [Hollerith,1889] descrisse, perfezionò e brevettò una serie di macchine che impiegavano le schede - sulle quali erano codificate informazioni come sesso, età, stato civile, analfabetismo - mediante fori praticati in posizioni standardizzate [Cfr. Hollerith, 1889, in Randell,1975, pp.129-130]. I dati codificati sulle schede non venivano solo letti, ma anche ordinati, smistati e organizzati dalla macchina, che era capace anche di eseguire semplici calcoli sui dati registrati.

Il macchinario di Hollerith si aggiudicò la gara per il censimento nazionale del 1890, dove processò con successo oltre 56 milioni di schede. Fu una sorta di *Big Data profiling* ante litteram. Hollerith ottenne svariati riconoscimenti, tra cui un dottorato (alla Columbia University nel 1890), per il suo sistema, che fu poi impiegato in diversi paesi. Nel 1896 fondò la *Tabulating Machine Company*, che nel 1911 si fuse con altre aziende dando origine alla *Computing-Tabulating-Recording Company*. Sotto la guida di Thomas J. Watson (1874 - 1956) questa compagnia fu ribattezzata *International Business Machine*, l'IBM.

Un rivale per il censimento del 1890, James Powers (1871-1927), sviluppò un sistema meccanico alternativo, tuttavia privo della flessibilità delle macchine di Hollerith, poiché evitava deliberatamente l'uso dell'elettricità per non infrangere i brevetti. Powers fondò una propria azienda, che fu acquisita nel 1927 dalla Remington Rand. IBM e Remington Rand dominarono per decenni il settore del calcolo e della contabilità automatizzata. [Randell, 1975, pp.125-127; Ceruzzi, 2012, pp. 6-9].

“Jacquard utilizzava le schede per il controllo, mentre Hollerith le usava per l'archiviazione dei dati. In seguito, anche gli impianti a schede perforate della IBM avrebbero utilizzato le schede per il controllo” [Ceruzzi, 2012, p. 8 trad. mia]⁴⁰⁶.

Proprio Presper Eckert, uno dei padri di ENIAC, pubblicò nel 1940 un volume intitolato *Punched Card Methods in Scientific Computation*, [Eckert, 1940] in cui illustrava l'uso del sistema a schede perforate per calcoli scientifici complessi. Qui Presper Eckert descriveva l'impiego di macchine standard di tabulazione e moltiplicazione elettrica, sottolineandone i vantaggi in termini di flessibilità e le possibilità di automazione delle operazioni calcolo, che evitava il dover ricorrere a sistemi “carta e matita”. “Controllo, memoria, calcolo, l'uso di circuiti elettrici

⁴⁰⁶ “Jacquard's and Hollerith's systems: Jacquard used cards for control, whereas Hollerith used them for storage of data. Eventually IBM's punched card installations would also use the cards for control”.

o elettronici: queste caratteristiche, quando combinate, costituiscono un computer” [Ceruzzi, 2012, p.9 trad. mia]⁴⁰⁷. Il design di ENIAC incorporò in sé questi elementi.

Oltre alle schede perforate, su cui erano immagazzinate 80 cifre nei patterns di buchi [Haigh, Ceruzzi, 2021, p.13] - per i sistemi di input-output, ENIAC era dotato di *display* [fig. 18]. Come abbiamo notato, alcune delle quasi 18000 valvole della macchina erano accoppiate per formare un circuito flip-flop. 10 flip-flop formavano un registro decimale circolante per sommare e memorizzare numeri. Dieci di questi contatori, ognuno rappresentante una cifra da 0 a 9, costituivano un accumulatore.

Sulla parte frontale dell’accumulatore, corrispondente a ciascun flip-flop, erano montate 102 lampade al neon sopra il livello degli occhi, grazie alle quali era possibile leggere il numero attualmente memorizzato. Due lampade al neon indicavano il segno positivo o negativo; le restanti 10×10 lampade (lette verticalmente) rappresentavano le dieci cifre di un numero a dieci cifre (lette orizzontalmente). Quando l’ENIAC eseguiva un calcolo, le lampade al neon nel campo quadrato si accendevano come palline da ping-pong, creando schemi in continuo cambiamento. [Thielmann, 2019, p.105 trad. mia]⁴⁰⁸

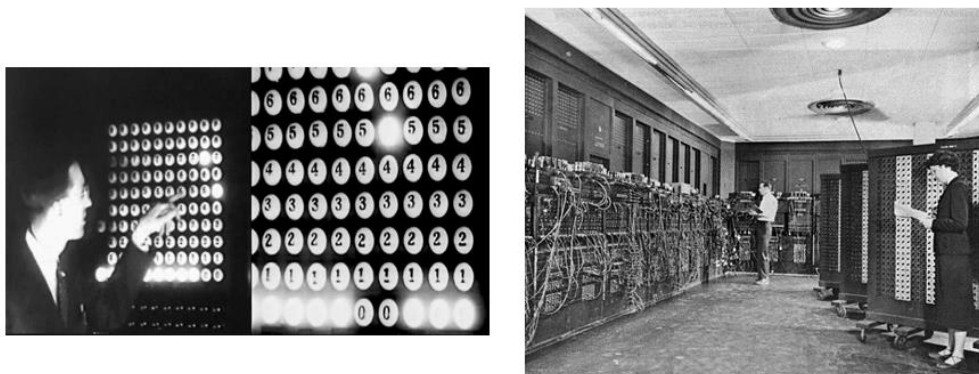


Fig. 18, Il display di ENIAC, in Thielmann, 2019; nell’immagine accanto, l’ENIAC.

⁴⁰⁷ “Control, storage, calculation, the use of electrical or electronic circuits: these attributes, when combined, make a computer”.

⁴⁰⁸ “On the front of the accumulator, corresponding to each flip-flop, 102 glow lamps filled with neon were attached above eye level, based on which the current stored number could be read off. Two neon bulbs signified the positive or negative sign; the remaining 10×10 glow lamps (read vertically) represented the ten digits of a (read horizontally) ten-digit number. When the ENIAC carried out a calculation, the neon bulbs positioned in the square field lit up ping-pong balls, yielding constantly changing patterns”.

Nelle conferenze stampa del febbraio '46 in cui l'ENIAC fu reso noto prima agli specialisti e poi al grande pubblico, furono proprio queste luci, posizionate in modo maggiormente percepibile, insieme al cablaggio visibile – che dava un effetto quasi organico – e la velocità delle operazioni di calcolo dimostrative⁴⁰⁹ a far parlare i media della macchina come “*Giant Brain*” [Cfr. Thielmann, 2019, pp.107-108]⁴¹⁰.

I pattern di luce dei display costituivano una rappresentazione sintattica, non semantica – e in questo il display di ENIAC si distingue dai moderni monitor dei computer – dei valori numerici elaborati: davano forma percepibile non all'informazione primaria, ma ai processi computazionali che la macchina stava eseguendo [Cfr. Thielmann, 2019, p. 107]. “Questi neon offrono uno dei controlli visivi più importanti sul funzionamento del computer poiché [...] forniscono un quadro quasi completo dei numeri presenti nella macchina e delle operazioni in corso” [Goldstine, Goldstine, 1946, trad. mia]⁴¹¹. A essere *visibile* sul display digitale e discreto⁴¹², erano i processi, fino prima *invisibili*, che avvenivano nella macchina. Era necessario che l'operatore possedesse una conoscenza della logica elaborata dai circuiti.

“L'ENIAC venne presentato come capace di calcolare la traiettoria di un proiettile sparato da un cannone più velocemente di quanto il proiettile stesso impiegasse a percorrerla” [Ceruzzi, 2012, p.46 trad. mia]⁴¹³ - impiegava circa 30 secondi per calcolarla [Haigh, Ceruzzi, 2021, p.13] ma, al contempo, “se un essere umano voleva interpretare la situazione di calcolo, l'osservatore doveva essere in grado di leggere i numeri dalle cifre lampeggianti e possedere

⁴⁰⁹ L'articolo di Kennedy del *New York Times* del 15 febbraio del 1946 sulla conferenza stampa del giorno prima da una efficace descrizione delle dimostrazioni di velocità di ENIAC “ “Guardate attentamente, potreste perderselo,” disse, [Arthur Burks] mentre si premeva un pulsante per moltiplicare 97.367 per sé stesso 5.000 volte. La maggior parte degli spettatori se lo perse: l'operazione avvenne in meno di un batter d'occhio. Per dimostrare l'estrema velocità dell'ENIAC, il dottor Burks rallentò poi l'azione di un fattore 1.000 e risolse lo stesso problema. Se i visitatori fossero stati disposti ad aspettare 162-3 minuti, avrebbero potuto osservare la risposta alla luce del mezzogiorno. Il passo successivo fu una moltiplicazione: 13.975 per 13.975. In un lampo apparve il risultato-195.300.625. Una tabella di quadrati e cubi di numeri fu generata in un decimo di secondo. Subito dopo, una simile con seni e coseni. Il lavoro fu completato e stampato su un grande foglio prima che la maggior parte dei visitatori potesse spostarsi da una stanza all'altra” [*New York Times*, 15 febbraio 1946]

⁴¹⁰ La rappresentazione mediatica dei calcolatori, dagli anni '40, assunse forme diverse a seconda dei contesti: la stampa popolare, come il *New York Times*, tendeva a presentare sia le macchine analogiche sia quelle digitali come “cervelli elettronici”, senza accentuare differenze sostanziali; la stampa scientifica, e in particolare riviste come *Scientific American*, iniziò invece a raccontare la storia dei computer come uno sviluppo orientato in modo crescente verso il digitale. Questa divergenza di narrazione ebbe un ruolo importante nel plasmare la percezione pubblica verso i soli sviluppi *digitali*. Cfr. Kline, 2019. pp.33-35.

⁴¹¹ These neons provide one of the most important visual checks on the operation of the computer since [...] they provide an almost complete picture of the numbers in the machine and the operations being performed.

⁴¹² Paper rappr. Continua discreta

⁴¹³ The ENIAC was touted as being able to calculate the trajectory of a shell fired from a cannon faster than the shell itself travelled.

conoscenze tabellari sulla costruzione numerica della traiettoria di un proiettile” [Thielmann, 2019, p.113 trad. mia]⁴¹⁴.

Le operazioni che potremmo definire di comunicazione con la macchina - dall'inserimento delle schede perforate alla lettura dei display, fino all'impostazione manuale dei commutatori e al collegamento di migliaia di cavi, in un lavoro che poteva richiedere ore per ogni singolo nuovo problema, la *programmazione* dell'ENIAC - erano affidate a sei donne che lavoravano a turni e che svolsero un ruolo fondamentale nello sviluppo e nel funzionamento dell'ENIAC: Kathleen McNulty Mauchly Antonelli (1921–2006), Frances Elizabeth “Betty” Snyder Holberton (1917–2001), Betty Jean Jennings Bartik (1924–2011), Marlyn Wescoff Meltzer (1922–2008), Ruth Lichterman Teitelbaum (1924–1986) e Frances Bilas Spence (1922–2012).

Queste sei programmatrici dovettero sviluppare - anche in maniera creativa (ad esempio, ricostruendo connessioni cablate alterate accidentalmente dal personale delle pulizie che, passando con le scope, staccava e reinseriva i cavi a caso) - una conoscenza completa del design e del modo di funzionare della macchina [Fritz, 1996]. Tra loro e la macchina, affinché le operazioni fossero eseguite con successo, doveva intercorrere una relazione *simbiotica* [Cfr. Thielmann, 2019, p.113]⁴¹⁵.

Accanto a queste donne, vanno ricordate le centinaia assunte dalla Moore School per calcolare le traiettorie delle tabelle di tiro, lavorando sei giorni alla settimana su due turni, in un piccolo edificio del campus dell'Università di Philadelphia, e le addette al cablaggio “*blue-collar*” su cui non si è soffermata l'attenzione storiografica e critica - a causa della tendenza del mondo digitale a squalificare il lavoro operaio e a non considerare i lavoratori, o le lavoratrici manuali, come *role model* positivi - senza le quali, tuttavia, ENIAC non sarebbe stato possibile [Haigh et. al, 2016, pp.71-74].

ENIAC era stato pensato per accelerare quantitativamente la produzione di tabelle di tiro. Le contingenze della guerra fecero sì che il collaudo della macchina avvenisse su dei calcoli relativi a un dispositivo bellico dalla potenza mai sperimentata prima: la bomba all'idrogeno, su cui spingeva fortemente il fisico Edward Teller (1908–2003), uno dei cosiddetti “marziani” di origine ungherese - insieme, tra gli altri, a John von Neumann e al matematico premio Nobel Eugene Wigner (1902–1995) - giunti negli Stati Uniti dalla terra magiara dopo l'ascesa del

⁴¹⁴ If a human wanted to interpret the calculative situation, the observer had to have the ability to read off numbers from the flashing digits and possess tabular knowledge on the numerical construction of the course of a projectile.

⁴¹⁵ In the case of the ENIAC, these were six female computers who worked in shifts with the computer in an almost symbiotic relationship.

nazismo e rapidamente assurti ai vertici della gerarchia scientifica americana [Bhattacharya, 2021]. Un vero e proprio battesimo del fuoco!

La *Super*, come veniva chiamata questa nuova bomba, era concepita per essere immensamente più potente delle bombe a fissione sganciate su Hiroshima e Nagasaki il 6 e l'8 agosto 1945. I calcoli relativi a tale ordigno non potevano avere – per evidenti motivi – alcuna verifica sperimentale preliminare. Il problema era individuare quale isotopo dell'idrogeno potesse effettivamente innescare una reazione di fusione auto-alimentata, alimentata da una piccola bomba atomica usata come detonatore. Il trizio, un isotopo rarissimo, poteva essere “innescato” in una reazione di fusione da una tale bomba; il deuterio, un isotopo relativamente comune, si credeva si potesse innescare solo a una temperatura irraggiungibile di 400 milioni di gradi. Il problema allora cercare di calcolare quale fosse la quantità di trizio per ottenere una miscela innescabile [Cfr. Haigh et al., 2016, pp.78-79]

Due fisici di Los Alamos, Nicholas Metropolis (1915–1999) e Stanley Frankel (1919–1978), erano alla ricerca di un dispositivo adatto per eseguire l'enorme massa di calcoli legata al problema della bomba all'idrogeno. Nel febbraio del 1945, Metropolis si rivolse a Paul Gillon per verificare le possibilità di “affitto” dell'ENIAC. Dopo una seconda visita nell'estate dello stesso anno, nel dicembre del 1945, sotto la direzione di Herman Goldstine, le programmatrici della macchina caricarono il problema sul calcolatore. Il programma implementato corrispondeva a circa un milione di schede perforate inviate da Los Alamos [Haigh et al., 2016, p.80; Dyson, 2012, p. 89, trad. it.].

L'ENIAC aveva a disposizione soltanto “meno 700 bit di memoria dati ad alta velocità” e “150 bit di memoria veloce” erano destinate al “controllo del programma” per quelle parti di programmazione che dovevano essere modificate rapidamente [Mauchly, 1980, pp. 547-548 trad. mia]⁴¹⁶. Il problema di ENIAC erano relativi alla memoria, non alla velocità (le conferenze stampa di presentazione daranno pubblica dimostrazione delle capacità della macchina da questo punto di vista. Le schede perforate rappresentavano un sistema di *data storage* per le computazioni intermedie del sistema, “ma si trattava di un sistema lento e soggetto ad errori” [Dyson, 2012, p.89 trad. It.].

⁴¹⁶ “Regardless of how many tubes may have been used, let us simply state that there were less than 700 bits of high-speed data storage in the ENIAC. On the other hand, there were 20 decades of iteration counters and 10 “ring-of-six” program stepper switches in the master programmer, and each accumulator had one decade for a program counter in its distributed control. Roughly, then, we could estimate about 150 bits of fast electronic storage were available for “program control”- that is, for storage of those parts of the program that had to be altered quickly”.

Le soluzioni ai problemi strutturali dell'ENIAC non tarderanno ad arrivare. Anzi, la loro teorizzazione sarà addirittura anticipata rispetto alla realizzazione materiale del calcolatore nato alla Moore School. Possiamo, per un momento, far nostra quella suggestiva narrazione delle origini dell'informatica impiegata da George Dyson [Dyson, 2012], che ha interpretato la nascita dei calcolatori come un vero e proprio *"mito della creazione"*. In questo racconto, l'ENIAC assume il ruolo che nel Nuovo Testamento fu di Giovanni Battista, il precursore. Ciò avviene anche a scapito di altre macchine precedenti o coeve – come Colossus, la macchina di Atanasoff, quella proposta da Wiener a Bush, il Mark I di Harvard⁴¹⁷ sviluppato da Howard Aiken e IBM sin dalla metà degli anni '30, o lo Z3 di Konrad Zuse in Germania⁴¹⁸ – tutte capaci di anticipare alcune caratteristiche fondamentali rispetto a ENIAC. A rappresentare il ruolo di Gesù di Nazareth in questa narrazione è il computer che implementa lo *stored-program principle* [Cfr. Haigh et al., 2016 pp.7-8].

Uno dei profeti della venuta di questo messia a programma memorizzato, John von Neumann, malgrado il suo ruolo di punta nei progetti relativi allo sviluppo di ordigni atomici,

⁴¹⁷ L'Harvard Mark I, come divenne noto l'IBM Automatic Sequence Controlled Calculator, fu progettato dal matematico e ufficiale della Marina Howard Hathaway Aiken (1900–1973) a partire dal 1935, e fu svelata al pubblico il 7 agosto 1944 presso l'Università di Harvard. Costruita in collaborazione con IBM, la macchina era lunga oltre 15 metri e alta circa 2,5 metri, composta da circa 760.000 componenti. Pur essendo digitale, il suo funzionamento era interamente elettromeccanico, basato su relè, motori e camme meccaniche. Poteva eseguire una moltiplicazione in circa 6 secondi. Il Mark I non era programmabile nel senso moderno: funzionava tramite nastri perforati che imponevano una sequenza rigida di istruzioni. La macchina ebbe tre evoluzioni successive: il Mark II (1947), il Mark III (1951), parzialmente elettronico, e infine il Mark IV (1952), completamente elettronico. Alla macchina collaborò la pioniera dell'informatica e dei linguaggi di programmazione Grace Murray Hopper (1906–1992). A Grace Hopper si deve l'introduzione del termine *bug* nel linguaggio informatico. Durante una sessione di lavoro sul Mark II, la macchina si arrestò improvvisamente: aprendo i relè, il team trovò una falena di grandi dimensioni incastrata e bruciata tra i contatti. Hopper la rimosse, la incollò nel registro di bordo [log book] con del nastro adesivo. Fu il primo caso documentato di *bug* [Beyer, 2009]. Sul Mark I si vedano Aiken, 1937; Aiken, Hopper, 1946; Cohen, 2003. Sulla figura di Aiken si veda Cohen, 1999. Su Grace Hopper si rimanda a Beyer, 2009.

⁴¹⁸ Konrad Zuse (1910–1995), ingegnere tedesco, costruì nel 1941 a Berlino lo Z3, considerato il primo calcolatore elettromeccanico funzionante, automatico e programmabile. Basato su circa 2.600 relè e sulla logica aritmetica binaria, operava con numeri in virgola mobile e leggeva istruzioni da nastro perforato [Ceruzzi, 2012, pp. 22–23; Bauer, 1980, pp. 503–524]. Era in grado di eseguire sequenze predeterminate di operazioni, ma non supportava istruzioni condizionali: era quindi solo potenzialmente Turing-completo, sebbene Zuse non avesse letto, all'epoca della costruzione della macchina, *On Computable Numbers* [Rojas, 1998]. Lo Z4, versione migliorata dello Z3, fu completato nel 1945 e venduto nel 1950 al Politecnico di Zurigo, dove rimase in uso fino al 1955. Tra il 1942 e il 1945, Zuse elaborò come argomento della sua tesi di dottorato il *Plankalkül*, un linguaggio formale di programmazione ad alto livello che prevedeva variabili, strutture dati, istruzioni condizionali, cicli e subroutine [Knuth, Pardo, 1980, pp. 202–211]. Zuse teorizzò inoltre l'idea, dalle forti implicazioni ontologiche, dello "Spazio calcolante" (*Rechnender Raum*) [Zuse, 1977], secondo cui l'intero universo può essere considerato come un automa computazionale finito, in cui ogni processo fisico corrisponde a un calcolo eseguito da una rete discreta di celle. In questa visione, la realtà fisica stessa coincide con la struttura computazionale dello spazio. Un'idea affine è stata proposta nel 1989 dal fisico John Archibald Wheeler (1911–2008), con il principio dell'*"it from bit"*: ogni ente fisico ("it") trae origine da un atto di osservazione, cioè da una scelta binaria sì/no, un "bit" di informazione. La realtà fisica, dunque, emergerebbe dall'informazione discreta, e non viceversa. Al fondo ultimo della realtà vi sarebbe allora, per Wheeler, l'informazione stessa [Wheeler, 1989]. Su Zuse si rimanda anche alla sua autobiografia [Zuse, 1983].

venne a sapere dell'ENIAC soltanto per caso, mentre aspettava il treno che lo avrebbe riportato a casa dopo una riunione del BRL ad Aberdeen nell'estate del 1944 [Bhattacharya, 2021]. Goldstine vide il celebre matematico ungherese sul marciapiede della stazione e decise di approcciarlo per parlargli. Il disinteresse iniziale di von Neumann verso gli argomenti della conversazione mutò di colpo quando seppe che stavano costruendo una macchina capace di eseguire 300 moltiplicazioni al secondo. "Quel momento" secondo Goldstine, "cambiò la sua vita per sempre" [Cfr. Dyson, 2012, p.87 trad. it.].

John von Neumann, in quanto membro del comitato di ricerca scientifica del BRL ebbe pieno accesso ad ENIAC. Ricorda Presper Eckert che durante la prima visita alla macchina nel settembre '44, von Neumann si interessò subito dell'architettura logica del calcolatore. "Se avesse chiesto subito cose come "Quanto è veloce?", saremmo rimasti delusi. Poiché invece chiese informazioni sulla logica di controllo, si creò subito un'intesa" [Eckert, 1980, p.525 trad. mia]⁴¹⁹.

L'interesse di von Neumann sulla logica del calcolatore si incrociò subito con i tentativi di Mauchly e Presper Eckert di costruire una macchina che superasse i problemi strutturali di ENIAC e implementasse il principio del programma memorizzato.

All'inizio del '45 sotto la supervisione di Goldstine, gli appunti di von Neumann su questa nuova macchina furono battuti a macchina e distribuiti (nel giugno dello stesso anno) alla Moore School. Questo testo, che, per rimanere nella metafora del mito della creazione che abbiamo utilizzato, che annunciava l'avvento del computer moderno era *First Draft of a Report on Edvac* [von Neumann, 1945].

Nel frattempo, sull'altra sponda dell'Atlantico, Alan Turing cercava di dare materialità alle sue idee intorno alla macchina universale, prima, tra mille difficoltà, al *National Physical Laboratory* e poi all'Università di Manchester.

Si dischiudevano le porte di un intero universo: dai 700 bit di memoria di cui era dotato ENIAC (una foto fatta da un odierno smartphone pesa dai 2 ai 6 megabyte, cioè è dalle 25.000 alle 60.000 volte più grande della memoria veloce del calcolatore della Moore School) sino ai 180-200 zettabyte che si stima rappresentino oggi la grandezza dell'*Infosfera* che indagheremo più avanti nello specifico. Per dare un'immagine concreta di questa colossale quantità di dati digitali, si consideri che un DVD contiene circa 4,7 gigabyte di dati, 200 ZB corrispondono a i

⁴¹⁹ "If he had first asked questions like "How fast does it work?" we would have been disappointed. Because he asked about the control logic, there was an immediate rapport".

44.600 miliardi di dischi. Impilandoli si otterrebbe una colonna alta oltre 53.520.000 chilometri, sufficiente a coprire la distanza tra la Terra e la Luna per più di 139 volte.

3. Turing in hardware: ACE e la Macchina di Manchester

A guerra appena conclusa, in Gran Bretagna, gli eroi *invisibili* di Bletchley Park ebbero un destino molto diverso da quello dei loro omologhi statunitensi. Tom Flowers ritornò a Dollis Hill pieno di amarezza. Le sue idee pionieristiche sull'elettronica, non supportate dal riconoscimento pubblico su quanto aveva realizzato durante il conflitto mondiale, gli avevano fatto guadagnare la reputazione di "pretenzioso" [Copeland, 2012, p.170]. Aveva visto ENIAC ad ostilità appena concluse e non ne aveva una grande considerazione: "Era solo un macinatore di numeri [*number cruncher*] - Colossus, con le sue sofisticate funzionalità per le operazioni logiche, era "molto più un computer dell'ENIAC" [Copeland, 2006f, p.107 trad. mia]⁴²⁰.

Turing, invece, il 1 ottobre 1945 iniziò il suo impiego presso il *National Physical Laboratory* (NPL) [Copeland, 2005b, p.40]. Il prestigioso ente scientifico britannico, sito a Teddington nei pressi di Londra, la cui prima riunione ufficiale si tenne il 16 maggio 1899, era nato dalle battaglie per il finanziamento della scienza nella seconda metà dell'Ottocento, sostenute dalle riforme del governo liberale di Gladstone - tra cui una riforma per garantire uno standard minimo di istruzione - e dall'*endorsement* della neonata, il 1869, rivista *Nature* [Magnello, 2005, pp.15-17]. Fu proprio l'autorevole rivista scientifica a dare notizia il 7 aprile che l'NPL avrebbe "esteso le sue attività con l'istituzione di una Divisione di Matematica" [*Nature*, 7 aprile 1945, n.155, p.431 in Croarken, 2005, p.23 trad. mia]⁴²¹.

A capo della neonata sezione fu nominato un energico uomo proveniente dallo Yorkshire, il matematico John R. Womersley [1907-1958]. Questi - che si considerava un erede di Babbage - era stato mandato negli Stati Uniti all'inizio del 1945 e aveva saputo dei piani intorno a ENIAC e al successore di questa macchina, l'EDVAC. Womersley aveva intuito il potenziale della computazione digitale. Nel futuro che immaginava "la computazione scientifica non sarebbe più stata eseguita dalle braccia di impiegati in carne ed ossa e da macchinari automatici". Queste nuove macchine, in una prefigurazione delle possibili conseguenze

⁴²⁰ Flowers' view of the ENIAC, which he saw just after the war: it was just a number-cruncher-Colossus, with its elaborate facilities for logical operations, was 'much more of a computer than ENIAC'.

⁴²¹ In April 1945 the journal *Nature* announced that the National Physical Laboratory was to 'extend its activities by the establishment of a Mathematics Division'.

disruptive dell'automazione di cui si dibatte oggi, “avrebbero fatto gran parte del lavoro delle classi più basse” [Copeland, 2012, pp. 161-162 trad. it.].

Il futuro capo della sezione matematica del NPL aveva già nel 1944 informato l'*Executive Committee* dell'ente scientifico britannico sulle potenzialità del calcolo digitale e, in qualità di membro dell'*Interdepartmental Technical Committee*, aveva spinto nello stesso anno affinché il NPL si dotasse di una sezione di matematica con lo scopo di condurre ricerche sulle nuove macchine calcolatrici. [Copeland, 2005b, p.37]

Womersley aveva letto, prima dello scoppio del conflitto, *On Computable Numbers* e ne aveva subito colto le implicazioni. Sin da subito aveva abbozzato degli schemi per costruire un calcolatore che incarnasse le idee di Turing impiegando la tecnologia elettromeccanica degli impianti telefonici, abbandonando però il progetto, perché la macchina, impiegando questa tecnologia, sarebbe stata troppo lenta per essere efficace. [Cfr, Womersely, 1944 in Copeland, 2005b, pp.38-39].

Le conseguenze della guerra tuttavia aprirono nuove possibilità. Venuto al corrente di quanto realizzato a Bletchley e sulle possibilità della computazione digitale, ora poteva coronare il suo sogno di realizzare “*Turing in hardware*” [Womersley, 1944, in Copeland, 2005b, p.39]. Localizzato grazie all'intervento di Max Newman, Turing – che nel frattempo, sull'onda dell'esperienza di Colossus, si era talmente appassionato all'elettronica da tenere una serie di conferenze sul tema ad Hanslope Park, dove era impegnato nel progetto segreto *Delilah*⁴²² - ricevette da Womersley, dopo un invito a casa con annessa illustrazione dei progetti di EDVAC, la proposta di un incarico temporaneo “di Senior Scientific Officer e una paga di 800 sterline l'anno” [Hodges, 1985, p. 400, trad. it.]. Turing accettò.

Flowers e Colossus gli avevano mostrato che era possibile realizzare la sua macchina universale. La proposta di lavoro gli offriva contestualmente la possibilità di approfondire i suoi interessi intorno al rapporto tra neurofisiologia e computazione. “Ritornato a Hanslope, Turing disse a Don Bayley che andava a “fabbricare un cervello”” [Copeland, 2012, p.163 trad. it.].

⁴²² Il progetto *Delilah* era un progetto elettronico segreto, promosso da Jumbo Travis, capo di Turing a Bletchley Park, che volle destinarlo a nuove frontiere della crittografia una volta concluso con successo il lavoro su *Tunny*. Turing si dedicò al progetto insieme a Tommy Flowers a Hanslope Park, nel Buckinghamshire, allestendo un laboratorio in una baracca Nissen a pochi chilometri da Bletchley Park. Ispirato al sistema di cifratura vocale SIGSALY dei Bell Labs (su cui aveva lavorato Shannon) *Delilah* mirava alla realizzazione di un sistema portatile per la cifratura della voce. Come spiegato, “Il metodo *Delilah* codificava parole allo stesso modo con cui la macchina *Tunny* cifrava i testi scritti: *Tunny* sommava chiavi oscuranti alle parole scritte, *Delilah* sommava chiavi oscuranti alle parole dette” [Copeland, 2012, p. 143, trad. it.]. Turing si basò sulla tecnologia crittografica esistente, e il mascheramento di *Delilah* avveniva in modo pseudocasuale, impiegando un meccanismo a cinque rotori modellato su quello della macchina *Enigma*. Per approfondire il lavoro di Turing su *Delilah* e la sua vita ad Hanslope si vedano, Copeland, 2017b, pp.183-187 e Hodges, 1985, pp.350-378 e 450-451.

Il compito di Turing, che nella divisione matematica dell’NPL costituiva una sorta di “one-man section” [Carpenter, Doran, 2017 pp.225,] era dare vita alla macchina che Womersely aveva battezzato, in onore del suo idolo Babbage, *Analytical Computing Engine* (ACE).

Turing trascorse il resto del 1945 a lavorare al design della macchina, mentre Womersley - per il quale Turing nutriva ben poca stima ⁴²³, circostanza che contribuì al naufragio del progetto originario dell’ACE - si adoperava per promuovere l’iniziativa, assicurando ai vertici del NPL (presieduto, come ricordato, da Charles Darwin III, la cui gestione burocratica e miope fu un ulteriore fattore del mancato sviluppo della macchina secondo la concezione turinghiana) che essa avrebbe superato i risultati ottenuti dagli americani e che, come Darwin ribadiva ai suoi superiori governativi, avrebbe potuto risolvere ogni problema computazionale del Paese.

Nel marzo del 1946, di fronte al comitato esecutivo presieduto da Darwin, accompagnato da Womersely e dal fisico Douglas Hartree⁴²⁴, Turing espose i risultati del suo lavoro. Il rapporto, sul quale ci concentreremo, aveva il titolo di *Proposals for development in the Mathematics Division of an Automatic Computing Engine* (ACE) [Turing, 1945] e “fu la prima specifica relativamente completa di un calcolatore elettronico digitale a programma memorizzato” [Copeland, 2005a, p.2, trad. mia]⁴²⁵.

Turing aveva compreso che il problema centrale per ogni macchina computazionale era il rapporto tra velocità e memoria (che, come abbiamo visto, in ENIAC era conflittuale). *Proposed Electronic Calculator* inizia proprio con l’esplicitazione di questo problema. La macchina digitale che veniva proposta, al contrario di quelle del passato che operavano su calcoli molto piccoli, o di quelle analogiche che lavorano su grandezze, “sarà in grado di affrontare e risolvere problemi nella loro interezza”. Questo sarà possibile perché “invece di servirsi ripetutamente del lavoro umano per estrarre materiale dalla macchina e reinserirlo al momento giusto, di tutto questo si prenderà cura la macchina stessa” [Turing, 1945, p.29 trad.

⁴²³ La disistima di Turing verso Womersley - che, pur essendo un matematico, non possedeva quelle competenze scientifiche che Turing considerava l’unico vero criterio per stabilire l’autorità - è riassumibile nel modo in cui pronunciava il suo nome, storpiandolo in *Wormsley* (da *worm*, verme). Per questa e per altre piccole perfidie di Turing nei confronti del suo superiore all’NPL si veda Copeland, 2012, pp. 175–176 trad. it..

⁴²⁴ Douglas Hartree (1897–1958) fu un fisico e matematico britannico che diede contributi fondamentali allo sviluppo del calcolo automatico sia analogico che digitale. Realizzò una versione semplificata, la prima costruita fuori dagli USA, del *Differential Analyzer* di Vannevar Bush presso l’Università di Manchester nel 1935 [Cfr. Hartree, 1940; Goldstine, 1972 p.99]. Durante la Seconda guerra mondiale collaborò a progetti di calcolo negli Stati Uniti, venendo coinvolto nello sviluppo e nell’uso dell’ENIAC.

⁴²⁵ “His technical report ‘Proposed Electronic Calculator’, dating from the end of 1945 and containing his design for the Automatic Computing Engine (ACE), was the first relatively complete specification of an electronic stored-program digital computer”.

it.]. I vantaggi in termini di velocità sono notevolissimi. Una volta eliminato il “freno umano” che con i suoi limiti e la sua fallibilità vincola le prestazioni della macchina si può prevedere esempio che “la moltiplicazione di due numeri di dieci cifre sarà eseguita in 500 microsecondi. L’esecuzione è probabilmente, circa 20 000 volte più rapida rispetto alla normale velocità delle macchine calcolatrici” [Turing, 1945, p.30 trad. it.].

Ma per far sì che queste prestazioni siano raggiunte, e che la macchina faccia, a velocità enormemente maggiore, ciò che fa un essere umano, questa deve essere dotata di “disponibilità analoghe”. Tre elementi di queste “disponibilità” sono fondamentali: “Primo, la carta su cui il calcolatore umano riporta i risultati e scrive le sue elaborazioni intermedie provvisorie; secondo, le istruzioni relative a quali processi devono essere eseguiti (di solito chi calcola queste istruzioni le ha nella sua testa); terzo, le tavole delle funzioni usate dal calcolatore devono essere disponibili in forma appropriata alla macchina” [Turing, 1945, p.30 trad. it.].

La macchina deve avere in sé i passi da seguire per risolvere il compito, la possibilità di accedere ai risultati intermedi e le modalità di esecuzione. Il calcolatore cioè deve essere dotato di “immagazzinamento di informazione”, cioè di una “memoria meccanica” [Turing, 1945, p.30 trad. it.].

L’assonanza con le considerazioni fatte da Turing quando teorizzava la sua macchina universale sono evidenti. È lo stesso Turing ad esplicitarle direttamente nella *Lecture on the Automatic Computing Engine*, tenuta dal matematico inglese il 20 febbraio del 1947 alla *London Mathematical Society* [Turing, 1947]. Questa è “la prima conferenza pubblica a menzionare l’intelligenza dei calcolatori, offrendo un’impressionante anticipazione di un nuovo campo” [Copeland, 2004b, p.355 trad. mia]⁴²⁶.

Di fronte ai suoi colleghi matematici Turing, parlando della “macchina calcolatrice automatica che è attualmente in corso di progettazione presso il National Physics Laboratory” [Turing, 1947, p.63 trad. it.] mette in evidenza come “approntare una memoria adeguata sia la chiave di volta del problema del calcolatore digitale, e certamente se si vuole che i calcolatori mostrino qualche specie di vera intelligenza li si deve fornire di capacità molto più ampie di quelle ora disponibili” [Turing, 1947, p.70 trad. it.].

Se si tiene a mente il rapporto tra intelligenza e macchina di Turing analizzato in precedenza, il problema della memoria risulta decisivo perché la macchina calcolatrice possa realizzare concretamente una macchina universale. Questa, infatti, può simulare qualsiasi altra

⁴²⁶ In London in 1947 Turing gave what was, so far as is known, the earliest public lecture to mention computer intelligence, providing a breathtaking glimpse of a new world.

macchina purché le venga fornita una descrizione adeguata sul proprio nastro: è su questo nastro che si concentra l'intera complessità della macchina da simulare. La memoria, controparte effettiva del nastro della macchina teorica universale, diventa così l'elemento fondamentale per l'esecuzione di qualsiasi procedura computazionale. "Questa caratteristica ha una corrispondenza nelle macchine calcolatrici digitali come l'ACE, che sono in effetti versioni pratiche della macchina universale, dotate di un dispositivo centrale di equipaggiamento elettronico e di una vasta memoria" [Turing, 1947, p. 72, trad. it.].

L'ACE, proposto da Turing nel documento del 1945 e poi nelle sue fasi di costruzione, rappresentava a tutti gli effetti quel "Turing in hardware" che concretizzava la macchina universale descritta in *On Computable Numbers*. Sulla carta, quella macchina che spingeva per trasformarsi in metallo era nata. Nel *Proposed Electronic Calculator*, per affrontare il problema cruciale della memoria, Turing proponeva un sistema basato su linee di ritardo acustico al mercurio, ispirato alle soluzioni di Presper Eckert, uno degli ideatori dell'ENIAC⁴²⁷. I segnali viaggiavano all'interno di un tubo lungo circa un metro e mezzo, e Turing pensò anche di migliorarne le prestazioni e ridurre i costi riempiendolo con un miscuglio di alcol e acqua, chiamato da alcuni "gin" [Copeland, 2012, p.179, trad. it.].

I *bit* - gli 0 e gli 1- arrivavano da un'estremità del tubo sotto forma di impulsi elettrici, venivano convertiti in segnali sonori intermittenti che attraversavano il tubo nel mezzo fluido (alla fine si esclude l'opzione gin), per poi essere riconvertiti in impulsi elettrici all'estremità opposta e amplificati per compensare eventuali degradazioni. In linea teorica, questo sistema consentiva di immagazzinare i dati indefinitamente, facendoli circolare in modo continuo all'interno del tubo. In pratica, però, i circuiti a ritardo si rivelarono estremamente delicati: il mantenimento di una temperatura costante era essenziale per prevenire errori, tanto che si sperimentarono soluzioni di ogni tipo, persino l'impiego della paglia come isolante. Il sistema, inoltre, risultava altamente sensibile a interferenze esterne: bastava il passaggio di una

⁴²⁷ Il primo modello funzionante di *delay line* ultrasonica fu realizzato da William B. Shockley (1910–1989), che fu inseguito uno degli inventori del *transistor*, dei Bell Labs, utilizzando una miscela di acqua e glicole etilenico come fluido. Il dispositivo trasformava un impulso elettrico in un'onda sonora che viaggiava nel fluido e veniva poi riconvertita in segnale elettrico dopo un ritardo determinato dalla lunghezza del tubo. Presper Eckert perfezionò l'idea impiegando il mercurio. Rispetto all'ENIAC, dove ogni bit veniva immagazzinato in un *flip-flop* - una coppia di tubi a vuoto, uno per bit - in una linea di ritardo era possibile memorizzare 1.000 bit con soli 10 tubi, riducendo di cento volte il costo dell'immagazzinamento. L'ENIAC, con il suo sistema macchinoso, poteva conservare al massimo 20 parole; con la nuova tecnologia si pianificava una memoria di 2.000 parole. Cfr. Goldstine, 1972, pp.189-191.

motocicletta nelle vicinanze per generare disturbi tali da provocare l'elaborazione di bit errati [Cfr. Copeland, 2012, pp.179-180 trad. it.].

La memoria della macchina nella proposta era costituita da unità preposte temporanee di riferimento veloce (MT), circa 50, ciascuna con una capacità di 32 cifre binarie. A queste si aggiungevano le unità di memoria dinamica (MD), in numero compreso tra 50 e 500, con una capacità totale compresa tra 1.500 e 150.000 cifre binarie, o *word*, parola che fu impiegata per la prima volta in questo senso proprio da Turing. Queste unità svolgevano sia la funzione di registri sia quella di memoria principale, secondo un'architettura seriale e distribuita, priva di un accumulatore centrale. L'ACE progettato era dotato di un'unità di *controllo logico* (CL), responsabile dell'interpretazione ed esecuzione delle istruzioni, spesso inoltrate alla parte centrale *aritmetica* (CA), che eseguiva le operazioni matematiche fondamentali (esclusa la divisione) e le varie manipolazioni. Le due unità, tra cui le differenze erano sfumate, erano collegate tramite "alberi" di selezione, che richiedevano un numero di valvole superiore a quello impiegato da CL e CA stessi. Il funzionamento dell'intero sistema era sincronizzato da un orologio centrale [Turing, 1945, pp.30-31 trad. it.].

Le operazioni aritmetiche si eseguivano direttamente tra i registri. In questo modo, il calcolatore integrava memoria e registri in un'unica struttura dinamica. La macchina era progettata per funzionare a una frequenza di 1 MHz, con parole di 32 bit [cfr. Campbell-Kelly, 2017, p.215; Carpenter, Doran, 2017, p.225].

La distribuzione seriale della memoria causava un problema di latenza, cioè un'istruzione non poteva essere usata se non era emersa dalla linea di ritardo. Questo comportava un ritardo medio di circa 0,5 millisecondi. L'idea di Turing per eliminare questo ritardo di latenza, non presente nel *Proposed Electronic Calculator* ma sviluppata pochi mesi dopo, fu quella di collocare le istruzioni in modo che emergessero dalla linea di ritardo esattamente nel momento in cui erano necessarie, eliminando così il tempo di attesa. Ogni istruzione avrebbe indicato la posizione della successiva, e anche i numeri sarebbero stati disposti strategicamente per ridurre al minimo i ritardi. Questo approccio - che sebbene rendesse la programmazione più complessa, garantiva prestazioni nettamente superiori - è noto oggi come *optimum programming*. [Cfr. Campbell-Kelly, 2017, pp.225-226].

Il set di istruzioni della macchina proposta era estremamente ridotto, solo 11, e progettato per riflettere direttamente l'organizzazione dell'hardware. La filosofia di Turing riguardo al *design* era di minimizzare l'uso dell'*hardware*, specialmente in problemi complessi come il calcolo decimale, e lasciare che questa complessità sia eseguita dal *software*. L'idea di

lasciare la componente fisica del calcolatore quanto più snella possibile era un'anticipazione del moderno principio di architettura dei calcolatori noto come RISC (*Reduced Instruction Set Computer*) [Cfr. Copeland, 2005a, p.2; Carpenter, Doran, 2017, p.225].

Oltre alle parti indicate *Proposed Electronic Calculator* indicava ulteriori componenti della macchina come un organo di ingresso (OI) “per trasferire istruzioni e altro materiale nel calcolatore dal mondo esterno” [Turing, 1945, p.32 trad. it.] composto da una parte meccanica – un lettore di schede Hollerith – e una elettronica interna al calcolatore; un organo di uscita (OU), anch'esso parzialmente meccanico mediante un riproduttore di schede perforate e una parte elettronica incorporata; convertitori dal binario al digitale; un meccanismo di accensione e un sistema di controllo della temperatura [Cfr. Turing, 1945, p.32-33 trad. it.]. Copeland fa notare come la macchina progettata da Turing, secondo queste caratteristiche, “prevedeva una memoria ad alta velocità con una capacità approssimativamente pari a quella di un primo Apple Macintosh (enorme per gli standard dell'epoca)” [Copeland, 2005a, p.2 trad. mia]⁴²⁸.

Il rapporto di Turing conteneva, inoltre, esempi di programmazione (con relative istruzioni), una serie di problemi che la macchina poteva teoricamente risolvere [cfr. Turing, 1945, pp. 48–49 e pp. 53–56], e anche una stima dei costi. Il matematico inglese fu un pioniere anche in quella tendenza alla sottostima della dimensione economica, tipica di ogni progetto informatico contemporaneo, stimando per ACE un totale di 11.200 sterline [Turing, 1945, p. 62, trad. it.].

I lavori per la realizzazione della macchina descritta in *Proposed Electronic Calculator* seguirono un percorso estremamente travagliato, segnato da ritardi, intoppi burocratici, cattiva gestione e difficoltà tecniche e organizzative, al punto che Turing non vide mai concretizzata la macchina da lui ideata e presentata nel rapporto. In un primo momento, Womersley e Turing si rivolsero a Dollis Hill per collaborare con Flowers alla costruzione dell'hardware. All'inizio del 1946 G. W. Radley, il capo di Flowers, scrisse a Womersley per confermare l'accordo, e Flowers stesso era fiducioso di riuscire in breve tempo a realizzare un “ACE minimo” [Cfr. Copeland, 2005b, p. 60].

Ma ben presto Flowers, su pressione dello stesso Radley, si ritirò dal progetto, soverchiato dal lavoro di ricostruzione del sistema telefonico inglese devastato dal conflitto contro i tedeschi. Dollis Hill tuttavia prestò l'opera di due ingegneri che erano stati collaboratori

⁴²⁸ “Turing's design for the ACE had much in common with today's RISC (Reduced Instruction Set Computer) architectures and called for a high-speed memory of roughly the same capacity as an early Apple Macintosh computer (enormous by the standards of his day)”.

di Flowers su Colossus, Allen “Doc” Coombs (1911–1995) e Bill Chandler (1916–2006). I due tecnici, purtroppo, si impantanarono per mesi in difficoltà relative alla costruzione delle linee di ritardo necessarie alla memoria di ACE [Cfr. Copeland, 2012, pp. 180-181].

Fallito questo tentativo, l’NPL fece un’offerta per un *visiting* annuale all’ingegnere statunitense Harry Huskey (1916–2017), che aveva lavorato all’ENIAC e a cui era stata offerta la direzione del progetto EDVAC. Huskey accettò e iniziò il suo lavoro alla divisione matematica il 4 gennaio 1947. Insieme ai collaboratori di Turing, Mike Woodger (1923–vivente) e al matematico, futuro premio Turing nel 1970, Jim Wilkinson (1919–1986), Huskey iniziò a lavorare a una versione semplificata di ACE, la “versione H” o *Test Assembly*. Era una soluzione all’insegna del pragmatismo americano tipico di Huskey: “Il primo obiettivo di Huskey era eseguire un semplice programma memorizzato utilizzando un’attrezzatura assolutamente minima e una singola linea di ritardo a lunghezza intera” [Copeland, 2005b, p. 64, trad. mia]⁴²⁹.

Nel 1947, i laboratori dell’NPL stavano costruendo linee di ritardo al mercurio secondo le specifiche fornite da Huskey, insieme a un telaio principale (*mainframe*) e ad altre componenti elettroniche. La realizzazione della linea di ritardo e la progettazione dell’elettronica erano già in corso; le scelte tecniche fondamentali – come i tipi di valvole e l’architettura dei circuiti – erano state definite, e si stavano scrivendo programmi per testarle.

Turing mostrò nei confronti del *Test Assembly* un atteggiamento che oscillava tra l’indifferenza e l’ostilità. Alla scelta di Huskey di costruire un calcolatore più piccolo, sebbene capace di affrontare compiti matematicamente impegnativi, oppose caparbiamente la propria convinzione che si dovesse rimanere fedeli al design originale dell’ACE. Probabilmente fu un errore da parte sua. Turing, contestualmente, stava facendo pressione su Darwin affinché l’NPL si dotasse di una divisione di elettronica. Il *Test Assembly* - attorno al quale si facevano previsioni ottimistiche di completamento entro il novembre del 1947 - pur essendo distante dal suo progetto e dalla sua filosofia, rappresentava comunque una buona base di partenza per la realizzazione concreta delle sue idee [Cfr. Copeland 2012, pp. 183–185, trad. it.; Copeland, 2005b, pp. 63–64].

Darwin finalmente acconsentì, nell’ultima parte del ’47, alla creazione di una sezione di elettronica all’interno della divisione Radio dell’NPL. A capo della sezione fu posto Horace Augustus Thomas. Fu una scelta pessima. Thomas era specializzato in un tipo di elettronica completamente diversa, quella industriale [Wilkinson, 1980, p.106] e “aveva fiutato

⁴²⁹ “Huskey’s first goal was to run a simple stored program using an absolute minimum of equipment and a single full length delay line”.

l'opportunità di costruire un piccolo impero all'interno dell'NPL" [Copeland, 2012, p. 186, trad. it.]. Fu proprio Thomas a suggerire a Darwin quella che può essere considerata "la peggiore decisione amministrativa di tutta la saga dell'ACE", quella di fermare i lavori del Test Assembly [Copeland, 2005b, p.64 trad. mia]⁴³⁰.

Il morale della divisione matematica si disintegrò, assieme alle speranze di costruzione di una macchina di Turing in hardware. Huskey, disgustato, terminato il suo contratto, tornò in America. Thomas, non soddisfatto del danno fatto, dopo sei mesi lasciò l'NPL per andare a lavorare per una ditta che produceva sapone [Cfr. Copeland, 2012, p. 188, trad. it.].

Anche Turing, alla fine, abbandonò. Chiese un anno sabbatico a Darwin per tornare a Cambridge e approfondire gli aspetti teorici del suo lavoro. Tornò anche in America, Qui visitò una serie di laboratori impegnati a realizzare macchine secondo il principio del programma memorizzato, dove si convinse che le divisioni matematiche e ingegneristiche dovessero lavorare sinergicamente. Darwin acconsentì a concedere a Turing il periodo di pausa, in cui il matematico elaborò le idee presenti in *Intelligent Machinery* [Turing, 1948], a condizione che egli accettasse un tipicamente britannico *gentleman agreement*, che lo impegnava a lavorare all'NPL per altri due anni dopo il suo ritorno dal sabbatico [Cfr. Copeland, 2012, p. 187, trad. it.].

Di ritorno dagli States, Turing venne a sapere che "l'ACE non era più avanti di come l'aveva lasciata" [Copeland, 2012, p. 187, trad. it.]. Fu troppo. Turing decise di accettare l'offerta che Max Newman, ora all'Università di Manchester, gli aveva fatto per seguirlo al *Computing Machine Laboratory*. Darwin, venutolo a sapere, licenziò Turing su due piedi. Tra i vari gruppi in corsa per la costruzione del calcolatore universale, Manchester ora era in vantaggio [Copeland, 2012, p. 187, trad. it.].

Il team ACE dell'NPL, nel frattempo, era stato affidato a Wilkinson, il quale – collaborando con Morley Colebrook (1893–1954), successore di Thomas alla guida della sezione elettronica – impiegò il principio di stretta sinergia tra matematici e ingegneri, lo stesso che Turing aveva ammirato negli USA. Così resuscitò il Test Assembly, trasformandolo, con una buona dose di *maquillage* e diplomazia, nel "modello pilota di ACE". Il *Pilot ACE* eseguì il suo primo programma con successo il 10 maggio 1950. Nel febbraio del 1952 fu ritenuto abbastanza affidabile da essere smantellato e trasferito alla sezione matematica [Cfr. Copeland, 2005, pp. 69–70; Copeland, 2012, p. 188, trad. it.].

⁴³⁰ "Then, in what was one of the worst administrative decisions of the whole ACE saga, Darwin summarily stopped the work. 'Morale in the Mathematics Division collapsed'".

È opportuno evidenziare un'ulteriore innovazione intorno alle macchine in costruzione all'NPL. All'inizio del lavoro su ACE fu costruita, da Woodger e D. W. Davies (1924-2000) della sezione matematica, una macchina "per la visualizzazione delle operazioni binarie" [Woodger, 2005, p. 331 trad. mia]⁴³¹: l'*ACE Simulator*. Realizzata nell'inverno 1949-1950, fu presentata per la prima volta il 30 gennaio 1950 in occasione delle dimostrazioni giubilari del NPL alla Royal Society.

La macchina, costruita con relè e lampade del Post Office, era montata su un rack alto sei piedi (circa 1 metro e 80). Il fronte era occupato da cinque pannelli che mostravano i costituenti dell'ACE, collegabili tra loro tramite spine volanti [*wander plugs*]. Nella parte posteriore vi era un telaio che conteneva banchi di relè, un uniselettore, una valvola tiratron con potenziometro per il controllo della velocità, e un alimentatore per il funzionamento dei relè e delle lampade.

Lo scopo della macchina era offrire una rappresentazione visiva del flusso dei bit. Per distinguere tra stati stabili e stati transitori, si usavano lampade bianche per i bit statici e lampade verdi per indicare quelli in transito. Il tempo base era diviso in quattro fasi: nella seconda e terza le lampade bianche rimanevano accese mentre si accendevano quelle verdi; poi, quando le verdi si spegnevano, si accendevano le lampade bianche della posizione successiva, generando così l'effetto di scorrimento. ACE Simulator rendeva visibili operazioni fondamentali che avvenivano nella macchina come il trasferimento su una linea di ritardo, la lettura e scrittura dei dati in memoria e le operazioni di addizione e di riporto [Cfr. Woodger, 2005, pp. 331-332].

Questo dispositivo - oggi esposto accanto al Pilot presso lo *Science Museum* di Londra e concepito come una sorta di analogo del display dell'ENIAC - non fu l'unico progettato e realizzato da Woodger e Davies. Nel 1949, infatti, i due membri della sezione matematica dell'NPL costruirono una macchina che, a livello hardware, si ispirava alle idee di Turing sull'apprendimento e sulle caratteristiche del cervello, a conferma di come tali concetti abbiano lasciato un'impronta diretta nella progettazione stessa delle macchine che il matematico inglese intendeva realizzare [Cfr. Sprevak, 2017, pp.278-285; Numerico, 2005, pp. 173-192].

Questa macchina era il Modello Cibernetico [*Cybernetic Model*], una "simulazione hardware di sei neuroni booleani" [Copeland, Proudfoot, 2005, p.132 trad. mia]⁴³². Il modello era composto da sei unità autonome in alluminio, ciascuna con una lampadina in cima, collegate

⁴³¹ "During the early days of the ACE work, when digital processing was novel, a demonstration machine was built as an aid to the visualization of binary operations".

⁴³² "Their Cybernetic Model, constructed in 1949, was a hardware simulation of six Boolean neurons".

ad un'unità di controllo. Ogni unità rappresentava un bit (lampadina accesa = 1, spenta = 0). A ciascuna si poteva assegnare una funzione booleana, con ingressi provenienti da qualsiasi altra unità. Durante un ciclo in quattro fasi, le funzioni elaboravano i valori correnti per generare nuovi stati, che sostituivano i precedenti. La macchina poteva avanzare a impulsi manuali oppure funzionare in modo continuo [Cfr. Woodger, 2005, p.332].

I giornali britannici avevano cominciato a interessarsi all'ACE e al lavoro dell'NPL sin dal 1946, quando il *Daily Telegraph*, il 7 novembre di quell'anno, pubblicò un articolo intitolato "*Britain to Make a Radio Brain*" [Daily Telegraph, 7 novembre 1946]. Quest'enfasi mediatica spinse la televisione a visitare l'NPL dove si realizzava la macchina. Woodger propose, a scopo dimostrativo, di utilizzare il Modello Cibernetico in una trasmissione intitolata *How the Brain Works*. Il programma, trasmesso il 13 novembre 1950, apriva la serie delle *Reith Lectures* tenute dal noto biologo J. Z. Young (1907–1997). In quell'occasione, Young illustrò come un polpo potesse apprendere a non attaccare un granchio se, dopo l'esposizione di un piatto bianco, veniva associato uno stimolo doloroso, come una scossa elettrica. Woodger utilizzò il Modello Cibernetico per imitare e riprodurre questo processo⁴³³.

"Alla fine di gennaio 1951, Norbert Wiener e Howard Aiken visitarono il NPL per vedere il Pilot ACE. Aiken rimase particolarmente colpito dal Modello Cibernetico, affermando: "Questa è la via del futuro"" [Woodger, 2005, p.334 trad. mia]⁴³⁴. Tuttavia, il modello, ritenuto una distrazione, fu smantellato.

In relazione all'attenzione dei media è interessante notare alcuni titoli di giornale riguardanti l'ACE. Sempre il *Daily Telegraph* intitolava un suo pezzo del 31 gennaio del 1950 "*ACE may be the Fastest Brain*" [Daily Telegraph, 31 gennaio 1950]. Il *Times* invece, lo stesso anno, faceva riferimento alle prestazioni previste della macchina con il suo titolo "*Month's Work in a Minute*" [The Times, 30 novembre 1950]. Il titolo del *Telegraph* del 7 novembre 1946 aveva come sottotitolo "*ACE Superior to U. S. Model*" [Daily Telegraph, 7 novembre 1946], paragonando quanto si stava realizzando all'NPL con le macchine in America. Proprio nel confronto con gli americani - forse per le differenti filosofie di costruzione, forse per il vantaggio pratico derivato da progetti come ENIAC ed EDVAC, o forse ancora per la frustrazione di vedere il proprio lavoro promosso soltanto in paragone con i successi pubblicamente riconosciuti negli Stati Uniti, mentre i risultati conseguiti a Bletchley rimanevano coperti dal segreto - quello

⁴³³ Il processo è descritto in Woodger, 2005, pp.333-334.

⁴³⁴ At the end of January 1951, Norbert Wiener and Howard Aiken visited NPL to see the Pilot ACE. Aiken was particularly impressed by the Cybernetic Model, saying: "That is the way of the future."

stesso anno, in un'intervista rilasciata all'*Evening News* del 23 dicembre, Turing si sentì in dovere di riconoscere agli americani "il merito del lavoro da somaro sull'ACE" [*Evening News*, 23 dicembre 1946, cit. in Copeland, Proudfoot, 2005, p. 113, trad. mia]⁴³⁵.

Intanto le travagliate vicende di costruzione proseguivano. Anche l'inquieto Womersley aveva lasciato l'NPL per lavorare prima all'ospedale St. Bartholomew di Londra, poi negli USA, dove trovò la morte prematura a soli 50 anni. L'NPL, nel frattempo, prese contatti nel 1949 con la *English Electric Company* per produrre una versione commerciale del Pilot, battezzata, con umorismo inglese, DEUCE (*Digital Electronic Universal Computing Engine*). Questo calcolatore, realizzato e consegnato alla divisione matematica nel marzo 1955, fu enorme successo che plasmò l'industria informatica britannica e rimase in uso fino agli anni '70⁴³⁶.

Altri colossi imprenditoriali inglesi, come l'EMI o la Packard-Bell, costruirono calcolatori ispirati alla macchina dell'NPL. Coombs e Chandler realizzarono una versione supersegreta dell'ACE, il MOSAIC, impiegato dall'aeronautica britannica per scopi difensivi. Alla fine, l'NPL riuscì anche a implementare una versione del modello di ACE ispirata al design di Turing: il "Big ACE", operativo nel 1960 [Cfr. Copeland, 2005a, p. 5; Copeland, 2012, pp. 188–190, trad. it.]⁴³⁷.

Prima di proseguire con il lavoro di Turing a Manchester, è necessario indagare un aspetto ulteriore. In *Proposed Electronic Calculator*, Turing scriveva: "Il presente rapporto dà un quadro abbastanza completo del calcolatore proposto. Si raccomanda tuttavia di leggerlo in parallelo al rapporto di J. von Neumann sull'EDVAC" [Turing, 1945, p. 31, trad. it.].

È impossibile delineare con assoluta certezza - a causa della segretezza che, sebbene con tempi, modalità e peso diversi, ha coperto il lavoro dei due matematici - il rapporto di reciproca influenza intellettuale tra von Neumann e Turing. Tuttavia, è possibile tracciare una ricostruzione.

Nel 1932, come ultimo dei libri acquistati grazie alla borsa di studio in memoria del suo compagno di scuola a Sherborne, Christopher Morcom, morto per tubercolosi bovina il 13 febbraio 1930 e con cui - insieme alla sua famiglia - Turing aveva un legame strettissimo, acquistò proprio *Die Mathematischen Grundlagen der Quantenmechanik* di von Neumann.

⁴³⁵ A newspaper report of 1946 stated that Turing 'gives credit for the donkey work on the A.C.E. to Americans'.

⁴³⁶ Si veda Campbell-Kelly, 2005, pp.162-170. Su Pilot ACE si veda anche Vowels, 2005, pp.223-263. Il punto di vista in prima persona di Huskey sul Test Assembly, Pilot ACE e Big ACE, insieme alla descrizione di Bendix G15, nell'opinione dell'autore, il primissimo personal computer, ispirato alla filosofia di Turing di sostituzione dell'hardware con il software, si trova in Huskey, 2005, pp.281-296. Un'altra testimonianza fornita da un protagonista della costruzione delle macchine dell'NPL e quella di Wilkinson, 1980, pp.101-114.

⁴³⁷ Un approfondimento su Big ACE si trova in Clayden 2005, pp.249-369. L'intero "albero genealogico" che realizza graficamente la discendenza del progetto ACE è in Campbell-Kelly, 2005, p.162.

Sebbene Turing rimanesse scettico sul ruolo esercitato dalla mente nell'interpretazione di Copenaghen, il libro lo aveva attratto e influenzato per "il modo con cui von Neumann si accostava al suo tema scientifico, e cioè usando per quanto possibile gli strumenti della logica" [Hodges, 1985, pp. 93 e 110, trad. it.].

È prpbabile che Turing abbia conosciuto von Neumann anche prima di recarsi nel 1936 a Princeton per il suo dottorato sotto Alonzo Church. Infatti, nell'estate del '35, il celebre matematico ungherese si era recato a Cambridge per tenere una serie di lezioni sul tema delle funzioni quasi periodiche, argomento sul quale si era cimentato anche Turing. "Alan certamente lo conobbe in quel periodo, e molto probabilmente proprio frequentando quel corso" [Hodges, 1985, p. 131, trad. it.; Cfr. Aspray, 1990, p.176].

Dopo che Turing aveva ottenuto il suo PhD negli USA, nel settembre del 1937, von Neumann scrisse una lettera in cui caldeggiava il suo nome per l'ottenimento della borsa Procter per l'Università di Princeton relativa all'anno accademico 1937-1938. Tuttavia, stranamente, nella lettera non si cita *On Computable Numbers* [Hodges, 1985, pp. 176-177].

Fu lo stesso von Neumann, nella primavera del 1938, a offrire a Turing un posto come suo assistente allo IAS, con uno stipendio di 1500 dollari l'anno. Turing tuttavia declinò l'offerta, poiché al King's College di Cambridge gli era stata rinnovata la *fellowship* e, inoltre, il lavoro con von Neumann riguardava ambiti che esulavano dalla teoria della computabilità, come la fisica quantistica. È molto probabile che, proprio in quel periodo, von Neumann fosse venuto a conoscenza del celebre articolo del 1936 [Cfr. Hodges, 1985, pp. 193-194, trad. it.].

Stanley Frankel, che lavorò con von Neumann a Los Alamos, ricorda come il matematico ungherese avesse piena consapevolezza, già nel 1943-44, quando iniziò ad occuparsi di calcolatori, dell'importanza dell'articolo di Turing del 1936, e che avesse insistito affinché lo studiasse. Sempre Frankel afferma: "Per come la vedo io, il ruolo essenziale di von Neumann fu nel far conoscere al mondo questi concetti fondamentali introdotti da Turing e il lavoro di sviluppo portato a termine nella scuola di Moore e altrove. [von Neumann] con fermezza fece notare a me, e ad altri ne sono sicuro, che fu Turing a concepirne i fondamenti" [Frankel, lettera a Randell 1972, in ed. Meltzer, Michie, 1972, cit. e trad. it. in Copeland, 2012, p. 172, trad. it.].

Sono numerosi i contesti in cui il matematico ungherese menziona con approvazione il lavoro di Turing: si pensi, ad esempio, alla conferenza del settembre 1948, sponsorizzata dall'Hixon Symposium, in cui von Neumann presentò la sua teoria degli automi [von Neumann, 1951]. In una lettera a Wiener del 26 novembre 1946, riflettendo sull'appropriatezza - su cui

torneremo - della metafora cibernetica applicata ai calcolatori e agli automi computazionali, von Neumann cita a sua volta in termini molto positivi il contributo di Turing:

Nonostante la sua complessità formidabile, questo soggetto ha prodotto risultati molto interessanti sotto la spinta degli sforzi di Pitts e McCulloch, Pitts [sic], Wiener e Rosenblueth. Il nostro pensiero - o quantomeno il mio - su tutto il tema degli automi sarebbe molto più confuso di quanto non sia, se questi sforzi estremamente audaci - *ai quali vorrei affiancare l'alquanto non-neurologica tesi di R. Turing* - non fossero stati compiuti [von Neumann, lettera a Wiener, 26 novembre 1946, trad. mia enfasi mia]⁴³⁸.

D'altra parte, anche il lavoro di von Neumann su EDVAC ebbe un impatto sul rapporto presentato da Turing all'NPL per il progetto ACE: "Per rappresentare le porte logiche dell'EDVAC, von Neumann aveva utilizzato una versione modificata di una notazione diagrammatica introdotta da McCulloch e Pitts in relazione alle reti neurali. Turing adottò tale notazione modificata e, in effetti, la estese notevolmente" [Copeland, Proudfoot, 2005, p.113 trad. mia]⁴³⁹.

Una cosa, comunque, è certa. Nel 1948, quando Turing abbandonò l'NPL, egli e "John von Neumann erano, forse, le sole due persone al mondo a comprendere pienamente l'idea di Turing che la macchina universale possa simulare il comportamento di una vastissima gamma di altri tipi di macchine, incluse le reti di neuroni sintetici" [Copeland, 2012, p. 273, trad. it.]⁴⁴⁰.

Manchester, assicuratasi Turing si era avvicinata a grandi passi alla realizzazione di questa macchina.

A Manchester, nel polveroso e freddo nord-ovest dell'Inghilterra, nell'estate del 1948, intanto, si era compiuta un'impresa. Un calcolatore bambino, la *Small Scale Experimental Machine* (SSEM), o, come verrà soprannominato, il *Manchester Baby*, era venuto al mondo grazie al lavoro di due ingegneri che provenivano dal *Telecommunications Research Establishment*

⁴³⁸ "Our thinking - or at any rate mine - on the entire subject of automata would be much more muddled than it is, if these extremely bold efforts - with which I would like to put on one par the very un-neurological thesis of R. Turing - had not been made. Yet, I think that these successes should not blind us to the difficulties of the subject, difficulties which, I think, stand out now just as - if not more - forbiddingly as ever".

⁴³⁹ "To what extent was the content of 'Proposed Electronic Calculator' influenced by the 'First Draft'? The former follows von Neumann's terminology and notation to some extent. This decision was a sensible one in the circumstances making it likely that Turing's report would be more readily understood. In order to depict the EDVAC's logic gates, von Neumann had used a modified version of a diagrammatic notation introduced by McCulloch and Pitts in connection with neural nets".

⁴⁴⁰ Un confronto puntuale e aggiornato della reciproca influenza tra Turing e von Neumann è in Copeland, Fan, 2023.

(TRE), Frederic C. Williams (1911–1977) e Thomas Kilburn (1921–2001), dove si erano occupati di tecnologia radar.

Nel settembre del 1945 si trasferì all'ateneo cittadino Max Newman, già professore di Turing a Cambridge e direttore della Newmanry a Bletchley Park durante la guerra, per assumere la cattedra Fielden di matematica [Copeland, 2011a, p.4].

Qui, Newman fondò, con i finanziamenti della Royal Society, il *Computing Machine Laboratory*. Il centro, nonostante il nome altisonante, occupava una sola stanza e dava l'idea di un "gabinetto pubblico" [Cfr, Williams, 1975, p.328; Copeland, 2012, pp. 201–202, trad. it.]. Si lascia all'immaginazione del lettore il contrasto con il contesto in cui si lavorava a Princeton, dove Newman si era recato nel 1946 per osservare i lavori sui calcolatori in costruzione negli Stati Uniti. Nonostante ciò, l'ex capo della Newmanry aveva posto le basi per la costituzione di un centro di ricerca sui calcolatori alternativo sia al polo americano BRL/IAS/Moore School, sia a quello inglese dell'NPL. La traiettoria che avrebbe portato il computer fino all'assolata California, e infine nelle mani di ciascuno di noi, trovò nella fredda e umida città del nord-ovest inglese un ulteriore snodo cruciale.

Williams fece domanda per una cattedra di Elettrotecnica all'Università di Manchester, che ottenne alla fine del 1946. Nella commissione sedeva anche Newman, il quale ascoltò con grande interesse le idee di Williams su un tipo di archiviazione della memoria a cui lavorava già negli anni al TRE. Pur avendo più volte dichiarato la propria ignoranza in materia di calcolatori⁴⁴¹, Williams era mosso da un "grande desiderio di trovare applicazioni pacifiche per quel patrimonio di competenze in campo elettronico che la guerra aveva prodotto attraverso le ricerche sui radar" [Hodges, 1985, pp. 442–443, trad. it.] e che aveva attirato anche le attenzioni del NPL [Cfr. Croarken, 1993, p.12]. "La prospettiva di unire le forze fu di certo vista come un matrimonio concepito in paradiso" [Copeland, 2012, p. 204, trad. it.]⁴⁴². I due uomini furono alla

⁴⁴¹ "Tom Kilburn ed io non sapevamo nulla di computer, ma molto di circuiti" [Tom Kilburn and I knew nothing about computers, but a lot about circuits. Williams, 1975, p.328 trad. mia].

⁴⁴² La Royal Society decise di assegnare a Newman un finanziamento di 35.000 sterline, articolato in 3.000 sterline annue per cinque anni destinate al personale e in 20.000 sterline, sempre su base quinquennale, per la realizzazione di una macchina destinata alla "ricerca fondamentale" [Napper, 2010; Hodges, 1985, p. 444, trad. it.]. Nella bozza di proposta alla Royal Society, redatta per uso interno dell'Università di Manchester, Newman - da poco rientrato dagli Stati Uniti, dove era stato presso l'Institute for Advanced Study di Princeton e aveva ricevuto persino un'offerta da von Neumann [Cfr. Dyson, 2012, p. 299, trad. it.] - dopo aver illustrato il problema della memoria per il progetto di macchina, citava esplicitamente Williams. Ciò potrebbe apparire singolare, poiché, reduce dall'esperienza statunitense, Newman era tornato con l'intenzione di impiegare il Selectron; inizialmente, infatti, non utilizzò i fondi della Royal Society per finanziare Williams, che disponeva ancora di quelli del TRE. Tuttavia, nell'autunno del 1947, resosi conto delle difficoltà legate al Selectron, Newman si rivolse a Williams e Kilburn [Cfr. Newman, 1948; Napper, 2010]. Il ruolo di Newman e la collaborazione del *Computing Machine Laboratory* con il Dipartimento di Ingegneria Elettrotecnica di Williams e Kilburn sono al centro di una piccola

base del fenomeno dei calcolatori di Manchester⁴⁴³. “Ognuno a suo modo, Newman e Williams crearono le condizioni perché a Manchester potesse iniziare una ricerca innovativa nel campo dei computer” [Croarken, 1993, p.9 trad.mia]⁴⁴⁴.

Il dispositivo di memoria in questione, che diverrà il cuore del Baby, era “il tubo a raggi catodici ad accesso casuale ad alta velocità” , il “tubo Williams” [fig.19] [Copeland, 2011a tra.mia]⁴⁴⁵.

Questo tubo - probabilmente ispirato dalle visite negli USA di Williams al Rad Lab, ai Bell Labs e alla Moore School tra il 1945 e i 1945 [Cfr. Croarken, 1993, p.12] – immagazzinava l'informazione, sotto forma di impulsi, in un componente relativamente disponibile e poco costoso, il tubo catodico impiegato per la tecnologia televisiva. Il funzionamento del tubo di basava sul fenomeno chiamato emissione secondaria che avviene nei tubi catodici. Quando un raggio di elettroni colpisce la superficie del tubo, ricoperta di fosforo, questa si illumina. Ma se l'energia è abbastanza alta, provoca anche il distacco di alcuni elettroni della sostanza che ricopre il tubo. Questi elettroni si muovono per un breve tratto e poi ricadono poco più in là sulla superficie del tubo. Il fenomeno crea, nel punto colpito, una carica positiva dove gli elettroni mancano, e una negativa nella zona circostante in cui ricadono. Il risultato è una piccola traccia elettrica, che permane per un tempo brevissimo e che può essere usata per immagazzinare l'informazione. [Cfr. Huskey, 2003].

Ricorda Williams, “Quando lasciai il TRE e mi trasferii a Manchester alla fine del 1946, si era raggiunto il punto in cui un tubo a raggi catodici (CRT) poteva memorizzare un solo bit” [Williams, 1975, p.327 trad. mia]⁴⁴⁶.

Il sistema alternativo al tubo Williams, la memoria di ritardo [fig.20], come abbiamo visto, implicava il problema che abbiamo definito di latenza. Bisognava, cioè, attendere che

controversia storiografica sugli sviluppi della computazione britannica. Copeland osserva come la narrazione “ortodossa” della storia dei calcolatori di Manchester tenda a considerare le due entità come inizialmente del tutto indipendenti e impegnate su progetti distinti. In realtà, vi fu una reciproca influenza fin dall'inizio, come dimostra, ad esempio, il fatto che il design logico del Baby fosse identico a quello che si stava sviluppando a Princeton [Cfr. Copeland, 2011a]. È da notare come, presente nella commissione della Royal Society incaricata di decidere in merito al grant da destinare al progetto di Newman, Darwin del NPL espresse un voto contrario. Una nuova macchina minava il suo progetto di un calcolatore unico per tutta la Gran Bretagna, l'ACE, sotto il suo diretto controllo [Cfr. Hodges, 1985, pp.444-445 trad. it.].

⁴⁴³ Per un resoconto sulla prima attività di ideazione e implementazione di calcolatori a Manchester, si veda Lavington, 1980, pp.433-443.

⁴⁴⁴ “In their different ways, Newman and Williams created the circumstances for innovative computer research to begin at Manchester”.

⁴⁴⁵ “The high speed random access cathode-ray tube (CRT) store known as the Williams tube”.

⁴⁴⁶ When I left TRE and moved to Manchester in late 1946, it had reached the state where 1 c.r.t. could store 1 binary digit.

l'informazione emergesse dal mezzo. La tecnologia a tubo catodico non presentava questo inconveniente. Williams “sapeva che poteva memorizzare i dati del computer – gli zero e gli uno – sullo schermo di un televisore. La memoria sarebbe sembrata una immagine televisiva in bianco e nero consistente in nient’altro che righe si segnali lampeggianti sullo schermo” [Copeland, 2012, pp.196-197 trad.it].



Fig.19 il tubo catodico Williams (CRT)



fig.20: La delay-line memory

Il problema del CRT era la “rigenerazione” [Williams, 1975, p.327]. Cioè come far sì che l'informazione non si perdesse, dissolvendosi, dopo pochissimo tempo. Williams, forse ispirato da quanto appreso negli USA dal gruppo di Presper Eckert⁴⁴⁷ – risolse, nel 1946, il problema della memoria elettronica ideando l’“impulso di previsione” [*anticipation pulse*], cioè un metodo per mantenere stabile un segnale sui CRT. Il sistema consisteva nel far colpire ripetutamente lo schermo dal fascio elettronico prima che la carica svanisse, rigenerando così il bit. Quando riuscì a mantenere stabile un singolo bit sullo schermo, capì che si sarebbe potuto estendere il metodo per memorizzarne migliaia [Cfr. Copeland, 2011; Copeland, 2012, p.198 trad.it.].

Come afferma Goldstine, “uno dei più grandi meriti di Williams fu comprendere che, se un'informazione può essere conservata per tempi molto brevi, allora può essere conservata

⁴⁴⁷ Ognuno dei due fu sempre convinto che l'altro gli avesse rubato l'idea [Cfr. Copeland, 2012, pp.197-199 trad. it]. Ad ogni modo Williams e Kilburn depositarono le prime domande di brevetto nel Regno Unito tra dicembre 1946 e ottobre 1947, seguite da richieste negli Stati Uniti nel dicembre 1947 e nel maggio 1949.

indefinitamente". Il tubo Williams "fu forse il miglior dispositivo di memoria disponibile fino all'invenzione della memoria a nuclei magnetici" [Goldstine, 1972, p.248 trad. mia]⁴⁴⁸.

Il dispositivo di Williams può essere considerato il "nonno" delle moderne RAM [*Random Access Memory*], ossia le memorie ad accesso casuale. Ogni cella può essere letta o scritta direttamente, senza dover attraversare, o attendere, la sequenza ordinata dei dati. A differenza delle linee di ritardo, il tubo Williams permetteva di accedere istantaneamente a qualunque punto della superficie, riducendo la latenza e introducendo per la prima volta un vero accesso casuale all'informazione digitale. Le prime RAM si svilupparono successivamente nella tecnologia a nuclei magnetici, la *Magnetic Core Memory*, dominante fino agli anni Settanta; da allora si è passati alle memorie a semiconduttore, come le DRAM [*Dynamic Random Access Memory*] e le SRAM [*Static Random Access Memory*], fino ad arrivare oggi alle memorie flash [*Floating-gate Access Memory*] e ad altre forme di memoria non volatile ad alta densità, che seguono un trend di costante aumento delle prestazioni e diminuzione di costi e dimensioni [Cfr. Kim, Koh, 2004].

L'invenzione di Williams era il cuore del Baby. Nel singolo tubo impiegato erano contenute "sole 32 "parole", ciascuna composta da 31 cifre binarie [la 32esima serviva per la rigenerazione del segnale], per contenere istruzioni, dati e operazioni in corso" [Williams, Kilburn, 1948, p.387 trad. mia.]⁴⁴⁹. In questa capacità minima Williams e Kilburn avevano, lentamente, bit per bit, caricato le 17 istruzioni per eseguire il compito stabilito, e cioè trovare il fattore più alto di 2^{18} . Il calcolatore impiegò 52 minuti, testando 130000 numeri e impiegando dai 3 ai 5 milioni di operazioni [Williams, Kilburn, 1948, p.387]⁴⁵⁰.

Era il 21 giugno 1948. "Mai prima di allora un *hardware* elettronico aveva eseguito un programma memorizzato" [Copeland, 2012, p.208 trad. it.]. Questo primo calcolatore elettronico a programma memorizzato, "con solo i mezzi più rudimentali di input/output e una capacità di memoria estremamente limitata, la macchina difficilmente poteva essere considerata un computer pratico" [Randell, 1975, p.352 trad. mia]⁴⁵¹.

⁴⁴⁸ "One of Williams' greatest accomplishments was to understand that if information can be stored for very short times, it can be stored indefinitely [...] the so-called was to be perhaps the best memory device available until the invention of magnetic core".

⁴⁴⁹ "The capacity of the store is at present only 32 'words', each of 31 binary digits, to hold instructions, data and working".

⁴⁵⁰ Per il design e l'architettura del *Baby* si veda Copeland, 2011b. Allo stesso testo si rimanda anche per l'analisi dell'influenza esercitata dalle idee di Turing su Kilburn, in particolare riguardo ai criteri di centralizzazione e decentralizzazione, nonché per le somiglianze tra i principi delle macchine sviluppate a Princeton e quelli adottati nel calcolatore di Manchester.

⁴⁵¹ "With only the most rudimentary means of input/output and an extremely limited storage capacity the machine was hardly a practical computer".

Lo scopo della macchina era, come indica il nome, di natura sperimentale. “Fu costruita principalmente per testare la validità del principio di memorizzazione adottato e per acquisire esperienza con questo tipo di macchina prima di intraprendere la progettazione di una macchina a grandezza completa”. Tuttavia, sottolineano i due autori, questa macchina era *universale*, poteva “essere usata per risolvere qualsiasi problema che possa essere ridotto a un programma di istruzioni elementari; il programma può essere modificato senza alcun cambiamento nei circuiti meccanici o elettromeccanici” [Williams, Kilburn, 1948, p.387 trad. mia]⁴⁵².

L’idea di Turing aveva, in senso letterale – data la natura del dispositivo impiegato – visto la luce. Nella storia della computazione e del digitale, se “il risultato raggiunto da Turing nel 1936 [...] è paragonabile all’invenzione della ruota”, ciò che aveva conseguito Williams, con il suo tubo e con la creazione del Baby “equivale alla progettazione del primo asse funzionante” [Copeland, 2012, p. 199 trad. it.].

Possiamo lasciar descrivere alle parole dello stesso Williams le sensazioni provate nel vedere il programma di 17 istruzioni funzionare sulla sua macchina:

Quando fu costruita per la prima volta, un programma veniva inserito con grande fatica e poi si premeva l’interruttore di avvio. Immediatamente i puntini sul tubo di visualizzazione iniziavano una danza frenetica. Nei primi tentativi era una danza di morte, che non portava ad alcun risultato utile e, cosa ancora peggiore, non forniva alcun indizio su cosa non funzionasse. Ma un giorno si fermò e lì, splendente nel punto previsto, c’era la risposta attesa. Fu un momento indimenticabile. Era il luglio [*sic*] del 1948, e da allora nulla fu più come prima [Williams, 1975, p.330 trad. mia]⁴⁵³.

Il *Baby* e il lavoro svolto a Manchester furono all’origine di un altro primato. Sulla base del progetto di Williams e Kilburn, la compagnia ingegneristica Ferranti realizzò il primo computer commerciale al mondo, il Mark I. Il primo esemplare di questa macchina fu installato

⁴⁵² “The machine is purely experimental, and is on too small a scale to be of mathematical value. It was built primarily to test the soundness of the storage principle employed and to permit experience to be gained with this type of machine before embarking on the design of a full-size machine; However, apart from its small size, the machine is, in principle, ‘universal’ in the sense that it can be used to solve any problem that can be reduced to a programme of elementary instructions; the programme can be changed without any mechanical or electro-mechanical circuit changes”.

⁴⁵³ When first built, a program was laboriously inserted and the start switch pressed. Immediately the spots on the display tube entered a mad dance. In early trials it was a dance of death leading to no useful result, and what was even worse, without yielding any clue as to what was wrong. But one day it stopped and there, shining brightly in the expected place, was the expected answer, It was a moment to remember. This was in July 1948 and nothing was ever the same again.

nel febbraio 1951, anticipando di poche settimane l'UNIVAC I, lanciato negli Stati Uniti dalla neonata EMCC (*Eckert-Mauchly Computer Corporation*) nel marzo dello stesso anno [Copeland, 2011a, p.5].

Il Mark I fu anche il secondo calcolatore – o meglio, *calcolatrice*, come si usava dire all'epoca in Italia – ad essere assemblato e installato nel nostro Paese, dopo il CRC 102A del Politecnico di Milano. Inaugurato il 31 ottobre 1955 rimase in funzione fino al 1963. Il Ferranti Mark I fu installato presso l'*Istituto per le Applicazioni del Calcolo* "Mauro Picone" (IAC) del CNR, dove fu ribattezzato FINAC, una crasi tra Ferranti e INAC, la sigla dell'istituto [cfr. Bonfanti, 2004; Guerraggio et al., 2010]⁴⁵⁴.

Il Mark I, ufficialmente *Manchester Automatic Digital Machine* (MADM), rappresentava la macchina "a grandezza completa" che Williams, Kilburn, Newman e il nuovo vicedirettore del *Computing Machine Laboratory*, Alan Turing, stavano progettando. A Manchester Turing ritrovò, oltre al suo maestro di Cambridge, anche Jack Good, che in quel periodo, grazie a Newman, aveva ottenuto un dottorato presso l'ateneo manchesteriano e svolgeva una funzione fondamentale di collegamento tra il laboratorio diretto dall'ex capo della Newmanry e la divisione di ingegneria elettronica di Williams e Kilburn [Cfr. Copeland, 2011a, p. 4].

Posati gli occhi sul Baby, Turing ne comprese subito i difetti. Lo SSEM non aveva che una – eufemisticamente- rudimentale struttura di input/output. I bit andavano inseriti a mano e bisogna cercare di leggerli sullo schermo quando comparivano. Il matematico inglese cercò di ovviare a queste difficoltà, che rendevano il Baby pressoché inutile per la computazione, adottando lo schema di interfaccia mediante nastro per telescriventi di Colossus (la sezione ingegneristica ignorava gran parte di quanto realizzato a Bletchley) [Cfr. Copeland, 2012, p.221 trad. it.].

Newman, dal canto suo, escogitava problemi per testare la macchina, dalla memoria limitata, ma capace di eseguire moltiplicazioni. Scelse un problema già discusso a Bletchley: trovare dei numeri primi molto grandi. Questo era un enigma che affliggeva i matematici sin dall'ipotesi di Marino Mersenne del 1644, secondo cui solo alcuni numeri della forma $2^p - 1$ – dove p è un numero primo – sarebbero a loro volta numeri primi. Mersenne indicava, tra gli altri, gli esponenti 17, 19, 31, 67, 127 e 257 come quelli corretti. Nel tempo, tra fine Settecento e primo Novecento, matematici come (1707–1783), F. Lucas (1842–1891) e D. H. Lehmer (1905–1991) avevano cercato di verificarne la correttezza con mezzi sempre più sofisticati. Il

⁴⁵⁴ A Bonfanti, 2004 si rimanda anche per una panoramica degli esordi della computazione nel nostro paese.

metodo ideato da Lucas nell'Ottocento si adattava perfettamente a un calcolatore binario: si trattava di ripetere elevazioni a potenza e sottrazioni di resti, un'operazione lunga ma lineare. Newman presentò il problema a Kilburn. Lui e il suo staff, nel giugno 1949, riuscirono a stipare il programma nei quattro tubi a raggi catodici, lasciando spazio per lavorare fino a $p = 353$. La macchina verificò i risultati storici, ma non emerse alcun nuovo numero primo [Cfr. Hodges, 1985, p.517].

Il calcolatore che andava delineandosi presentava le seguenti caratteristiche tecniche: era un calcolatore binario a virgola fissa, la cui unità di controllo e calcolo - la CPU - si basava su tecnologia a valvole elettroniche. Utilizzava *word* da 40 bit, ciascuna contenente due istruzioni da 20 bit, e operava in sistema *backward binary*, in cui il bit meno significativo veniva memorizzato a sinistra. La memoria principale, ad accesso casuale, era costituita da tubi Williams e poteva contenere 256 parole; ad essa si affiancava una memoria secondaria su tamburo magnetico che ne ampliava la capacità fino a 16.000 parole. I registri interni, inclusi accumulatori e registri indice, erano anch'essi implementati su tubi Williams. Quanto alle prestazioni, il Mark I eseguiva operazioni aritmetiche elementari in poco più di un millisecondo, mentre una moltiplicazione richiedeva circa 2,16 millisecondi. Il suo ritmo interno era scandito da cicli regolari di circa 240 microsecondi [Cfr. Lavington, 1980, pp. 434-436].

Tra le mansioni di Turing come vicedirettore del *Computing Machine Laboratory* – che, nel frattempo, nel 1951, era stato eletto membro della Royal Society di Londra – vi era anche quella di redigere un manuale “di operazioni e programmi di base [...] chiamato *Programmer's Handbook* [Turing, 1951]” [Hodges, 1985, p. 519, trad. it.]. In quegli stessi anni, i rapporti tra Turing e Kilburn si erano deteriorati al punto che quest'ultimo arrivò a vietargli l'accesso alla macchina, per poi giungere a un compromesso che limitava l'uso del calcolatore da parte di Turing a due notti la settimana [Cfr. Copeland, 2012, p. 213, trad. it.].

La guida, dattiloscritta, conobbe successivamente numerose versioni migliorative. L'interfaccia, come detto, riprendeva i criteri già adottati in Colossus. L'input avveniva inizialmente tramite un lettore di nastro perforato a cinque tracce capace di leggere 250 caratteri al secondo, mentre l'output era inviato a un punzonatore di nastro e a una stampante [Cfr. Lavington, 1980, p. 436]. Inoltre vi erano i meccanismi di visualizzazione dei bit necessari a una macchina che impiegava la tecnologia di Williams a tubo catodico.

In questo contesto emerge un cambiamento nell'atteggiamento di Turing rispetto alla proposta originaria dell'ACE, in particolare per quanto riguarda la relazione tra macchina e operatore. Se ai tempi dell'NPL Turing sosteneva che dovesse essere la programmazione – e

non l'elettronica – ad adattarsi alle esigenze dell'utente, ora, lavorando su una macchina reale, in grado di incarnare quella visione, cambiò approccio. Nella conferenza del 1947 aveva affermato: “Noi siamo in grado di risolvere questi problemi senza modificare la macchina, grazie al solo lavoro con carta e matita, che alla fine si traduce nel fornire alla macchina le istruzioni appropriate” [Turing, 1947, p.75 trad. it.]

Nel *Programmer's Handbook*, invece, si raccomandava “fortemente” all'utente di imparare una tabella di conversione in base 32, poiché rifletteva il funzionamento del calcolatore, costruito intorno ai tubi Williams [Cfr. Turing, 1951, p. 3]. Tale tabella elencava i 32 simboli scelti per rappresentare le 32 cifre della macchina, ma rendeva più complessa la trascrizione nel codice Baudot. Lo stesso Turing se ne rese conto. Infatti nel suo manuale si può leggere: “Questi simboli sono essenzialmente il codice del telescrivente, eccetto che le combinazioni 00000, 01000, 00100, 00010, 11011, 11111, che nel vero telescrivente sono rappresentate da nessun effetto, avanzamento riga, spazio, ritorno carrello, cambio cifra, cambio lettera” [Turing, 1951, p. 3, trad. mia]⁴⁵⁵.

A Turing - una sorta di hacker *ante litteram*, che passava le notti, o almeno quelle concessagli da Kilburn, da solo di fronte ai pannelli di programmazione della macchina - “personalmente gli riusciva semplicissimo lavorare direttamente con un'aritmetica in base 32, che era quella con cui lavorava la macchina, e si aspettava lo stesso dagli altri” [Hodges, 1985, p. 218, trad. it.]. L'idea, implicita nel suo approccio, era che fosse l'utente ad adattarsi alla logica della macchina, non viceversa. Una filosofia diametralmente opposta a quella perseguita nello stesso periodo da Maurice Wilkes (1913–2010) a Cambridge con l'EDSAC, secondo il quale l'hardware doveva essere progettato per facilitare il più possibile la comprensione da parte dell'utente umano: “[...] veniva impiegata la lettera A come simbolo dell'addizione”⁴⁵⁶.

⁴⁵⁵ These symbols are essentially the teleprinter code, except that the combinations 00000, 01000, 00100, 00010, 11011, 11111 which in true teleprint are represented by no effect, line feed, space, carriage return, figure shift, letter shift.

⁴⁵⁶ Questi due approcci - quello di adattamento dell'utente alla macchina, volto a un suo potenziamento in relazione alle capacità del calcolatore, e quello che invece mira a rendere la macchina più accessibile all'utente - rispecchiano una tensione fondamentale della storia dell'interfaccia uomo-macchina (HCI). Come scrive E. R. Petrick, i due differenti approcci riflettono una tensione fondamentale che attraversa la storia dell'interazione uomo-macchina: “Sostengo che vi siano due temi principali che attraversano la storia dell'HCI: da un lato, la tensione tra *augmentation* e *user-friendliness* nel modo in cui il computer e l'utente comunicano tra loro; dall'altro, la tendenza verso interfacce sempre più invisibili nel tempo” [“I argue that there are two major themes running through the history of HCI: the tension between augmentation and user-friendliness in how the computer and the user communicate with each other and the trend toward increasingly invisible interfaces over time” Petrick, 2015 trad. mia]. Al primo filone si possono ascrivere le ricerche dell'*Augmentation Research Center* di Doug Engelbart presso lo Stanford Research Institute, sviluppate a partire dal celebre rapporto *Augmenting Human Intellect: A Conceptual Framework* (1962). Qui l'interfaccia è concepita come un *flusso di energia* e informazione: lo scopo dell'interazione era, attraverso circuiti di feedback che coinvolgono anche la dimensione corporea, aumentare le capacità

Il *Programmer's Handbook*, tuttavia, opera a un livello molto più – per usare un termine contemporaneo – *basic* rispetto alle raffinate proposte di programmazione contenute in *Proposed Electronic Calculator*. Questo probabilmente a causa della tecnologia impiegata nel Mark I, che si basava sull'uso dei tubi catodici Williams accoppiati ad una memoria magnetica. Tale architettura prevedeva un trasferimento continuo dei dati dal tamburo magnetico ai tubi Williams, con la conseguenza che ogni subroutine, una volta invocata, doveva essere ricollocata integralmente su una nuova pista magnetica.

Turing era consapevole del problema, ma lo liquida nel manuale con una certa dose di ironia, giudicandolo di scarsa rilevanza. Infatti, si può leggere:

Le subroutine di qualsiasi routine possono a loro volta avere subroutine. È come nel caso delle pulci grandi e piccole. Non sono certo del significato esatto che il poeta attribuiva all'espressione "e così ad infinitum", ma sono incline a pensare che intendesse che non si può assegnare un limite alla lunghezza di una catena parassitaria di pulci, piuttosto che credere effettivamente in catene infinitamente lunghe. Questo è certamente il caso delle subroutine: si arriva sempre, prima o poi, a una routine priva di subroutine [Turing, 1951, p.60 trad. mia]⁴⁵⁷.

intellettive e operative dell'utente [Engelbart, 1962; Bardini, 2000]. Al secondo approccio appartengono invece le ricerche di Don Norman sugli artefatti cognitivi, definiti come «quei dispositivi artificiali che conservano, mostrano o manipolano informazioni allo scopo di svolgere una funzione rappresentazionale e che influenzano la performance cognitiva umana» ["those artificial devices that maintain, display, or operate upon information in order to serve a representational function and that affect human cognitive performance" (Norman, 1991, trad. mia)]. Norman mette però in guardia dal considerarli meri strumenti di "amplificazione": gli artefatti non estendono direttamente la mente, ma trasformano la natura del compito, spostando parte dell'elaborazione su un supporto esterno e alleggerendo il carico cognitivo tramite *affordances*. Queste ultime, secondo Norman, sono le proprietà relazionali di un oggetto che rendono immediatamente percepibili all'utente le azioni possibili, guidandone il comportamento e riducendo il rischio di errore. L'obiettivo non è quindi potenziare la mente in senso astratto, ma ridisegnare i compiti in modo da facilitarne la soluzione e rendere più naturale l'interazione. Lo scopo di tali *affordances*, come Norman chiarisce già in *The Psychology of Everyday Things* [Norman, 1988], è proprio quello di rendere visibili e intuitive le possibilità d'uso incorporate negli artefatti, facilitando l'apprendimento e l'efficacia dell'interazione. Tali idee trovano ulteriore sviluppo in *The Invisible Computer* [Norman, 1998], dove Norman invoca una progettazione centrata sull'utente e sugli *information appliances*: dispositivi semplici e piacevoli da usare, in cui la tecnologia diventa "invisibile". In questa traiettoria si colloca anche la scansione storica proposta da Yvonne Rogers [2012], che distingue tre fasi nell'evoluzione dell'HCI. La fase *classica*, dagli anni '80 ai primi anni '90, è dominata dai modelli cognitivi, come quelli elaborati da Card, Moran e Newell in *The Psychology of Human-Computer Interaction* [Card et al., 1983], che si concentrano sulla capacità, quantificabile, dell'elaborazione cognitivo-informazionale dell'essere umano inteso come utente generico che interagisce con la macchina, con l'obiettivo di ottimizzare prestazioni, ridurre gli errori e misurare il carico cognitivo. La fase *moderna*, inaugurata dal lavoro di Lucy Suchman in *Plans and Situated Actions* [Suchman, 1987], mette in discussione questo impianto astratto, sottolineando invece la dimensione situata, sociale e contestuale dell'interazione: l'uso delle tecnologie è visto come parte integrante delle pratiche reali e non come mera esecuzione di piani. La fase *contemporanea*, dal 2000 in avanti, amplia ulteriormente l'orizzonte includendo emozioni, creatività, esperienza estetica e design critico, in continuità con la diffusione delle tecnologie ubiquitarie e mobili, e con una crescente attenzione ai contesti culturali e sociali.

⁴⁵⁷ "The subroutines of any routine may themselves have subroutines. This is like the case of the bigger and lesser fleas. I am not sure of the exact meaning the poet attached to the phrase 'and so ad infinitum', but am inclined to think that he meant there was no limit that one could assign to the length of a parasitic chain of fleas, rather than

La Guida per programmatori proponeva, inoltre, un'ulteriore, ennesima, innovazione, frutto della genialità del suo autore, che dimostra come persino un ammasso di acciaio, tubi, cavi e ingranaggi come la Macchina di Manchester potesse dar corpo a quell'universalità che Turing aveva intuito sin dal 1936.

Il calcolatore era stato dotato dagli ingegneri di una sorta di sirena, chiamata *hooter*, che emetteva un segnale sonoro quando collegata a specifici output della macchina o quando questa richiedeva l'attenzione degli operatori. Turing si rese conto che il suono sgradevole dell'*hooter* poteva essere sfruttato per generare melodie musicali, ed elaborò un'istruzione apposita, la *"/V"*. "Quando viene eseguita un'istruzione con il simbolo di funzione */V*, viene applicato un impulso al diaframma di un altoparlante. Ripetendo questa operazione in modo ritmico, si può produrre una nota continua, ricca di armoniche" [Turing, 1951, p. 24, trad. mia]⁴⁵⁸.

Le operazioni della macchina erano sincronizzate da un orologio centrale che, come un metronomo, scandiva il tempo in unità regolari: otto battiti costituivano un'unità temporale, e un ciclo completo durava 1,92 millisecondi, corrispondente a una frequenza di 521 cicli al secondo [Turing, 1951, p. 24]. Eseguendo la funzione */V*, un battito dell'orologio diventava sonoro, ma solo per l'istante brevissimo corrispondente all'emissione di un singolo impulso. Questi suoni, scrive Turing, "sono chiaramente udibili come qualcosa a metà tra un colpo secco, uno scatto e un tonfo" [Turing, 1951, p.24 trad. mia]⁴⁵⁹.

Turing comprese che quel suono - corrispondente al Do centrale - poteva, se ripetuto più volte e con opportune variazioni nei pattern delle istruzioni, produrre suoni diversi. Ad esempio, abbassando il tono di una quinta, "allora l'orecchio sente note musicali differenti [Cfr, Turing, 1951, p.24; Copeland, 2012, p.220 trad. it.]⁴⁶⁰.

La musica elettronica generata dal computer - oggi alla portata di chiunque scarichi la giusta app - è nata a Manchester. Qui la computazione non era soltanto diventata *visibile*, grazie al tubo Williams, ma era anche divenuta *udibile*, per merito della programmazione di Turing.

Intanto, nel 1951 si presentò a Manchester un ex compagno di studi di Turing al King's College, Christopher S. Strachey (1916–1975). Appartenente all'*upper class* britannica - suo zio

that he believed in infinitely long chains. This certainly is the case with subroutines. One always eventually comes down to a routine without subroutines".

⁴⁵⁸ When an instruction with function symbol */V* is obeyed an impulse is applied to the diaphragm of a loudspeaker. By doing this repeatedly and rhythmically a steady note, rich in harmonics, can be produced.

⁴⁵⁹ Single pulses applied to the loudspeaker are distinctly audible as something between a tap, a click, and a thump.

⁴⁶⁰ Sul funzionamento delle istruzioni musicali della Macchina di Manchester, si veda Copeland, Long, 2017, pp.233-248.

era lo scrittore Lytton Strachey, figura di spicco del gruppo di Bloomsbury - Strachey insegnava matematica e fisica nella prestigiosa scuola privata *Harrow*.

Strachey si era recentemente interessato ai calcolatori digitali, entrando inizialmente in contatto con il gruppo dell'NPL attraverso Mike Woodger. Venuto a conoscenza dei lavori in corso a Manchester, scrisse a Turing: i due condividevano un background simile, dalla frequentazione di Cambridge fino alle difficoltà personali legate all'orientamento sessuale nell'Inghilterra della prima metà del Novecento, e tra loro si instaurò subito un'intesa. Quando il suo ex compagno di studi arrivò a Manchester, Turing gli consegnò una copia del *Programmer's Handbook* - già all'epoca noto per la sua oscurità e difficoltà di lettura - per tenerlo occupato [Cfr. Copeland, 2012, pp. 221-223].

Strachey si applicò al manuale di Turing. Dopo l'estate tornò a Manchester e mostrò i frutti del suo lavoro sulla guida per programmatori. "Appassionato pianista, apprezzò il potenziale delle concise indicazioni su come programmare le note musicali" [Copeland, 2012, p.223 trad. it.].

Studiando il *Programmer's Handbook* aveva prodotto una ventina di pagine di istruzioni di programmazione. Ricevute indicazioni da Turing sul funzionamento operativo della macchina, il calcolatore fu lasciato a sua disposizione tutta la notte. Il mattino seguente Turing verificò che il programma scritto da Strachey funzionava. Dopo qualche attimo di silenzio, dagli hooter del computer, venne fuori l'inno nazionale britannico: *God Save the King*. "Qualche settimana dopo, Max Newman sentì il computer che stava "macinando" *God Save the King* e scrisse in fretta una lettera a Strachey suggerendogli che avrebbe potuto gradire un lavoro di programmazione nel laboratorio" [Copeland, Long, 2017, p.237 trad. mia]⁴⁶¹.

La notizia che il macchinario sapeva comporre musica si diffuse presto. Il rotocalco britannico *Courier and Advertiser* titolò il 28 febbraio 1952 "*Electronic Brain can sing now*" [*Courier and Advertiser*, 28 febbraio 1952]. La BBC si interessò subito alla cosa e inviò una troupe insieme ad Auntie, la popolare presentatrice del programma radio per bambini *Children's Hour*, per registrare la performance dell'enorme musicista meccanico. La macchina eseguì l'inno britannico, la filastrocca per bambini *Baa Baa Black Sheep* e una versione di *In the Mood* di Glenn Miller. Il calcolatore (ora si era arrivati alla seconda versione, il Mark II), pieno di glitch, si bloccò nel mezzo dell'esecuzione di quest'ultima. La presentatrice Auntie non si fece

⁴⁶¹ "A few weeks later Max Newman heard the computer grinding out God Save the King and quickly wrote a letter to Strachey suggesting he might like a programming job in the lab".

sfuggire l'occasione di rimarcare come la macchina non fosse nel giusto *mood*! [Cfr. Copeland, Long, 2017, p.237].

Il contributo di Strachey non si limitò a fare della macchina di Manchester l'antesignano della musica elettronica. Il distinto programmatore britannico "aggiunse altri elementi all'alone di fascino del computer, creando un programma che scriveva lettere d'amore" [Copeland, 2012, p.225 trad. it.]. Creò un paio di righe di istruzioni e le memorizzò nella macchina insieme a una lista di parole e di frasi prese dal *Thesaurus* di Roget e le accoppiò a una delle istruzioni del manuale di Turing, un programma di generazione stocastica di numeri casuali, la funzione chiamata "*The random numbers generator*" [Cfr. Copeland, 2012, p.225 trad. it.; Turing, 1951, pp.25-26].

I risultati, firmati M.U.C (Manchester University Campus), risultavano essere autentici in modo inquietante, del tipo:

CARO AMORE MIO

SEI L'EMOZIONE CHE MI ACCOMPAGNA. IL MIO AFFETTO CURIOSAMENTE SI AGGRAPPA ALLA TUA VOGLIA ARDENTE. IL MIO DESIDERIO ANELA IL TUO CUORE. SEI IL MIO MALINCONICO AFFETTO: IL MIO TENERO DESIDERIO

TUO CARNALMENTE

M.U.C. [Strachey, 1954, p.26 trad. it. in Copeland, 2012, p.226 trad. it.].

Turing, probabilmente romantico allo stesso modo, ma più pragmatico, usava il programma di Strachey per scrivere le sue lettere personali, come quella inviata al suo amico Robin Gandy a proposito il suo esame orale di dottorato:

CARO/ROBIN//////////SCUSA////È/VERAMENTE/IMPOSSIBILE/FARE/IL/TU
O/ESAME ORALE/PRIMA [Copeland, 2012, p.226 trad. it.].

Ad oggi, nel 2025, si inviano quasi 380 miliardi di messaggi di posta elettronica ogni giorno, mentre il mercato delle intelligenze artificiali generative, capaci di produrre testi di ogni tipo - persino lettere d'amore - è stimato globalmente intorno ai 135-140 miliardi di dollari [Stanford AI Index Report, 2025, p. 30]. Risalire la genealogia di questi fenomeni significa anche tornare alle piogge della Manchester degli anni '50.

Le innovazioni di Strachey non finiscono qui. Utilizzò i sistemi di visualizzazione dei bit della macchina – che normalmente servivano agli operatori per rendersi conto degli stati interni del calcolatore – per creare, con un programma che aveva scritto quando frequentava il gruppo NPL a Londra, una scacchiera per il gioco della dama, che anticipava il lavoro di Samuel [Samuel, 1959].

Dialogando con l'utilizzatore umano mediante il generatore di frasi, la macchina, dopo aver salutato il giocatore suggerendogli di leggere le istruzioni, invitava l'utente a fare le sue mosse. E se questo ci metteva troppo tempo, non era raro che potesse leggere frasi del tenore di: "MI RIFIUTO DI PERDERE ALTRO TEMPO, VA A GIOCARE CON UN ESSERE UMANO" [Copeland, 2012, p.228 trad. it.].

Il programma di Strachey simulava il gioco anticipando due mosse e valutando ciascuna posizione possibile secondo semplici regole euristiche (es. una dama vale tre pedine). Calcolava sistematicamente le conseguenze di ogni mossa, scegliendo quella che massimizzava la propria posizione. Il processo imitava il ragionamento umano, ma in modo meccanico e meticoloso. La velocità del computer era di circa una mossa ogni 10 secondi [Cfr. Copeland, 2012, p.229 trad. it.].

Il programma di gioco della dama implementato nel calcolatore di Manchester dava corpo alle idee di Turing sull'intelligenza delle macchine e sul rapporto con le "facoltà" umane, già esemplificate dal suo lavoro sugli scacchi e dal concetto di *child machines*. Che egli considerasse questo un compito fondamentale per testare le potenzialità di un calcolatore nell'apprendere – e nel mostrare segni di intelligenza – è testimoniato dal fatto che, ancora prima di assumere il suo ruolo al *Computing Machine Laboratory*, nel *Proposed Electronic Calculator* presentato ai vertici dell'NPL figurava, quasi sorprendentemente, tra i problemi che l'ACE avrebbe potuto affrontare, uno riguardante proprio il gioco degli scacchi, probabilmente influenzato dal lavoro simile di Shannon. Nel rapporto si legge: "Data una posizione in una partita a scacchi, la macchina potrebbe elencare tutte le combinazioni vincenti fino a una profondità di tre mosse da ambo i lati" [Turing, 1945, p. 56, trad. it.].

Riguardo al modello di ACE che desiderava costruire, Turing aggiungeva:

Può la macchina giocare a scacchi? Sarebbe abbastanza facile far sì che la macchina giochi partite piuttosto brutte; "brutte" perché gli scacchi richiedono intelligenza, e abbiamo affermato all'inizio di questo paragrafo che la macchina dovrebbe essere trattata come se mancasse completamente di intelligenza. Ci sono indicazioni, tuttavia, del fatto che è possibile far sì che la macchina manifesti

intelligenza al prezzo di farle correre il rischio di commettere ogni tanto gravi errori” [Turing, 1945, p. 56, trad. it.].

I risultati *in hardware* della macchina di Manchester aprivano dunque a Turing la possibilità concreta di realizzare quello che aveva sempre desiderato: costruire un cervello.

4. Un avvento cibernetico: EDVAC e il suo design

È possibile ora ritornare sull'altra sponda dell'Atlantico, dove avevamo lasciato John von Neumann, all'epoca consulente del Ballistic Research Laboratory, alla stazione di Aberdeen. Venuto a sapere quasi per caso da Goldstine dell'esistenza del calcolatore di Mauchly ed Eckert, von Neumann si rivolse a quest'ultimo chiedendogli chiarimenti sulla struttura logica della macchina. È lo stesso Goldstine a ricordare come, nelle discussioni alla Moore School con il nuovo gruppo impegnato sul progetto PX, i temi centrali fossero la flessibilità, la velocità di ordinamento, la codifica dei problemi, la capacità di memoria, la rapidità della macchina e il design dei circuiti [Cfr. Goldstine, 1972, p.187].

Nel 1944, ben prima che l'ENIAC fosse ultimato, i suoi ideatori stavano già pensando a una macchina che lo sostituisse. “Essi si resero conto che passare da un compito all'altro sull'ENIAC avrebbe richiesto periodo di riconfigurazione lungo e soggetto a errori, e che il suo metodo di controllo distribuito imponeva una massa confusa di *colli di bottiglia*” [Haigh, Ceruzzi, 2021, p.13 trad. mia enfasi mia]⁴⁶²

Goldstine ricorda come “già nel 1944 vi era una comprensione generale, all'interno del gruppo della Moore School, che gli ordini per una macchina digitale potessero essere memorizzati in forma numerica su nastro”. Ma all'ottimismo iniziale seguì subito la consapevolezza delle debolezze dell'ENIAC come macchina general purpose: “In particolare, io [...] ero infastidito dalla macchinosità dei meccanismi di programmazione dell'ENIAC e dal piccolo numero - 20 - di registri di memoria elettronici modificabili. Avevamo già immaginato l'uso dell'ENIAC per calcoli molto più elaborati rispetto alla semplice risoluzione di equazioni balistiche” [Goldstine, 1972, p.184 trad. mia]⁴⁶³

⁴⁶² “By mid-1944, more than a year before ENIAC was working, its creators were already working on ideas for a successor. They recognized that switching ENIAC between jobs would require a lengthy and error-prone period of reconfiguration and that its distributed control method imposed a confusing mass of bottlenecks limiting, for example, the number of branches that could be included in a program.

⁴⁶³ “Thus by 1944 there was general understanding among the group at the Moore School that the orders for a digital machine could be stored in numerical form on tape. [...] This revealed a number of serious problems and

Memoria; costi; dimensioni; programmabilità; affidabilità; architettura logica: questi erano i punti deboli dell'ENIAC. "Allo stesso tempo, alla Moore School si stava conducendo una rivalutazione generale delle caratteristiche operative dell'ENIAC. Questa mise in luce diversi problemi seri e carenze della macchina come strumento di calcolo scientifico general-purpose" [Goldstine, 1972, p. 184 trad. mia]⁴⁶⁴.

Goldstine, nell'agosto del 1944, avviò i contatti con i vertici del BRL affinché fosse stipulato un nuovo contratto governativo per una macchina migliorata rispetto all'ENIAC. Nel frattempo era comparso in scena von Neumann. È probabile che fu la sua entrata in campo a catalizzare e accelerare gli eventi. Il matematico ungherese era molto più esperto, e connesso ad alti livelli, nella formalizzazione e nella presentazione di progetti su larga scala rispetto al team di ingegneri della Moore School.

Il 29 agosto 1944, in una riunione del *Firing Table Reviewing Board*, alla presenza di von Neumann, Goldstine e dello staff del BRL, il consiglio raccomandò la costruzione di una nuova macchina che, superando le limitazioni dell'ENIAC, avrebbe accelerato la produzione di tabelle di tiro, aumentato la capacità computazionale per trattare l'intera gamma delle equazioni balistiche e supportato la ricerca di base. La stima economica del progetto PY – elaborata da Brainerd – fu di 105.600 dollari per un'attività della durata di dodici mesi. Oltre al nuovo apparato elettronico, su cui stava lavorando Presper Eckert (le linee di ritardo a mercurio), la proposta menzionava anche i tubi RCA che abbiamo analizzato [Cfr. Goldstine, 1972, p. 185-186; Haigh et al., 2016, p.134]. La speranza era "che uno dei problemi di onda d'urto di von Neumann potesse essere risolto in dieci secondi su ENIAC, rispetto alle quattro ore richieste da una macchina a schede perforate" [Haigh et al., 2016, p. 133 trad. mia]⁴⁶⁵.

Le riflessioni sui principi e sulle scelte progettuali del calcolatore destinato a succedere all'ENIAC furono raccolte da von Neumann in un documento che, se si accetta la narrazione mitica della nascita del calcolatore a programma memorizzato, può essere considerato il suo annuncio. Il testo, dattiloscritto sotto la sua supervisione, fu riprodotto con il ciclostile e

deficiencies in the machine as an instrument for general-purpose scientific calculation. In particular I [...] was bothered by the clumsiness of the mechanisms for programming the ENIAC and by the small number-20-of electronically alterable storage registers. We already had visualized doing much more elaborate calculating than solving ballistic equations"

⁴⁶⁴ "Thus by 1944 there was general understanding among the group at the Moore School that the orders for a digital machine could be stored in numerical form on tape. [...] This revealed a number of serious problems and deficiencies in the machine as an instrument for general-purpose scientific calculation".

⁴⁶⁵ "They estimated that one of von Neumann's blast-wave problems could be solved in ten seconds on ENIAC, as opposed to four hours on a punched-card machine".

distribuito alla Moore School il 30 giugno 1945, in un numero di copie limitato “a quante se ne poterono stampare prima che le ruote del ciclostile cedessero” [Dyson, 2012, p. 93, trad. it.].

Il documento, lungo 101 pagine, prese il titolo di *First Draft of a Report on the EDVAC* – acronimo di *Electronic Discrete Variable Automatic Calculator*. Sulle copie del *First Draft* – cui non seguì mai un secondo – compariva un solo nome, quello di John von Neumann, senza alcun riferimento a Mauchly o a Presper Eckert.

La paternità delle idee contenute nel *First Draft*, al pari del segreto imposto sui risultati di Bletchley, rappresenta una delle questioni più controverse della storia dello sviluppo dei calcolatori. È però indubbio che l'ingresso di von Neumann nei progetti della Moore School ne modificò profondamente la scala. Il matematico ungherese non portava soltanto la sua straordinaria capacità analitica, ma anche l'abilità di muoversi con disinvoltura tra le alte sfere dell'amministrazione scientifica. Il suo interesse per il nuovo calcolatore, espresso – e celato – attraverso riflessioni cibernetiche sulla logica computazionale, era in realtà orientato alla risoluzione di problemi matematici cruciali, come la soluzione di equazioni differenziali parziali a due o tre variabili [von Neumann, 1945, 1.1], direttamente connessi al suo coinvolgimento nel Progetto Manhattan, dei quali non poteva parlare apertamente. “Così come l'ENIAC era stato modellato sul problema delle tabelle di tiro, l'EDVAC fu modellato sulla dinamica dei fluidi della bomba atomica.” [Haigh et al., 2016, p.135 trad. mia]⁴⁶⁶.

L'EDVAC, il calcolatore delineato nel documento – in cui, significativamente, il nome della macchina non compare mai – può essere descritto, con la dovuta cautela nell'uso del concetto, come un triplice “salto paradigmatico” nel senso delineato da Thomas Kuhn nel suo classico testo *La struttura delle rivoluzioni scientifiche* [Kuhn, 1962; cfr. Haigh et al., 2016, pp. 142-143]. Il “paradigma EDVAC” è stato poi utilizzato retrospettivamente come categoria analitica nella storia dei calcolatori, talvolta anche a scapito di sviluppi precedenti, come quelli britannici esaminati in precedenza.

Relativizzando la portata storica del *First Draft* rispetto all'ENIAC e ai suoi limiti, l'EDVAC ne rappresenta il superamento in tre ambiti fondamentali: la struttura hardware, la programmabilità e l'architettura [Haigh et al., 2016, p. 143].

Per quanto riguarda il primo punto, il *First Draft* si apre dichiarando che intende delineare “la struttura di un sistema di calcolo digitale automatico ad altissima velocità, e in

⁴⁶⁶ “Just as ENIAC was shaped by the firing-table problem, EDVAC was shaped by the fluid dynamics of the atom bomb.

particolare il suo controllo logico” [von Neumann, 1945, 1.1 trad. mia]⁴⁶⁷. Il cuore di questo calcolatore, concepito come meno elefantiaco rispetto all’ENIAC, era imperniato sul sistema di valvole a linea di ritardo – già previsto prima del coinvolgimento di von Neumann con la Moore School – destinate a costituirne la memoria [von Neumann, 1945, 4.1 e 4.2]. A ciò si affiancava una logica interamente implementata in elettronica, basata sulla rappresentazione numerica binaria [von Neumann, 1945, 4.1-4.2; 5.1-5.2].

La memoria interna ad alta velocità proposta nel documento è nettamente superiore a quella dell’ENIAC. Nel loro primo report sull’EDVAC, *Automatic High-Speed Computing. A Progress Report on the EDVAC* [Mauchly, Eckert 1945]⁴⁶⁸ redatto nel 1945 da un punto di vista prevalentemente ingegneristico – a differenza della trattazione logica del *First Draft* – Mauchly ed Eckert scrivevano: “In effetti, l’EDVAC disporrà di una memoria interna ad alta velocità con una capacità di circa 2.000 numeri decimali di dieci cifre, mentre l’ENIAC ne può contenere solo 20. Ciononostante, si prevede che l’EDVAC richiederà solo un decimo dell’equipaggiamento dell’ENIAC” [Mauchly, Eckert, 1945, trad. mia]⁴⁶⁹.

Se la tecnologia impiegata era quella preesistente della *delay-line*, si può individuare il contributo di von Neumann nella proposta di una “una capacità di memoria esageratamente ampia. Il *First Draft* specificava una memoria di 8.000 parole, ciascuna lunga 32 bit, destinata principalmente alla memorizzazione di numeri piuttosto che di istruzioni” [Haigh et al. 2016, p.134 trad. mia]⁴⁷⁰.

Questa capacità di memoria interna è direttamente riconducibile ai bisogni computazionali del Progetto Manhattan, non affrontabili mediante la memoria esterna su schede perforate utilizzata dall’ENIAC. Lo osservano Mauchly ed Eckert quando scrivono: «Di conseguenza, la soluzione di ampie classi di equazioni differenziali parziali non lineari sarà facilmente gestibile con l’EDVAC, mentre tali problemi possono essere affrontati con l’ENIAC solo con l’ausilio di una capacità di memoria esterna molto più lenta, sotto forma di schede

⁴⁶⁷ “The considerations which follow deal with the structure of a very high speed automatic digital computing system, and in particular with its logical control”.

⁴⁶⁸ H. Huskey, lo stesso computer scientist che abbiamo incontrato nello sviluppo dell’ACE, ha redatto le parti 5,6, e 7 del rapporto. *Automatic High-Speed Computing. A Progress Report on the EDVAC*, inizialmente segreto, fu desecretato nel 1946, a guerra finita. Questo rapporto ebbe un ruolo cruciale sulla causa per i brevetti dell’ENIAC che vide protagonisti Mauchly e P.Eckert.

⁴⁶⁹ “In fact, the EDVAC will have an internal high-speed memory capacity for about 2,000 ten-digit decimal numbers, while the ENIAC accommodates only 20. Nevertheless, the EDVAC is expected to require only one-tenth as much equipment as the ENIAC”.

⁴⁷⁰ Von Neumann’s most substantial contribution in this area was probably to push the team toward what might have been considered an extravagantly large memory capacity. The *First Draft* specified a memory of 8,000 words, each 32 bits long, primarily intended for the storage of numbers rather than instructions.

perforate» [Mauchly, Eckert, 1945, trad. mia]⁴⁷¹. Lo sottolinea anche l'apertura del *First Draft*, laddove si afferma che “*Un sistema di calcolo automatico*” come quello descritto nel documento «è un dispositivo (solitamente altamente composito) in grado di eseguire istruzioni per svolgere calcoli di un notevole grado di complessità - ad esempio, la soluzione numerica di un'equazione differenziale parziale in 2 o 3 variabili indipendenti» [von Neumann, 1945, 1.2, enfasi originale]⁴⁷². Tali equazioni sono centrali nei problemi di idrodinamica, la cui risoluzione era necessaria per la realizzazione della bomba atomica, ed è proprio la macchina descritta nel *First Draft* a proporre metodi per la loro soluzione automatizzata [von Neumann, 1945, 12.3].

La possibilità di trattamento delle PDE, inseguita per lo meno sin dai calcolatori analogici di Bush, comincia a materializzarsi nelle pieghe elettroniche della memoria interna di un calcolatore universale in costruzione.

Per quanto riguarda il secondo punto, la *programmabilità*, l'EDVAC rappresentò un deciso passo in avanti rispetto all'ENIAC, che pur offrendo un certo grado di flessibilità operativa, rimaneva vincolato a una configurazione manuale laboriosa. Come ricorda Presper Eckert:

Nel gennaio del 1944, eravamo perfettamente consapevoli del fatto che l'ENIAC presentava una certa buona generalità, ma anche una certa scarsa generalità. Occorrevano ore per configurare la versione originale non modificata, ma era comunque meglio del Differential Analyzer di Bush, che di solito richiedeva giorni per essere impostato, usando cacciaviti, martelli di piombo e simili” [Eckert, 1980, p. 530, trad. mia].

L'ENIAC doveva infatti essere predisposto dalle sue programmatrici per ogni compito specifico attraverso un lungo e dispendioso lavoro di cablaggio e connessione di cavi e prese. Il *First Draft of a Report on the EDVAC*, invece, introduce un modello radicalmente nuovo, in cui le istruzioni - definite numericamente secondo una forma logico-algebrica - vengono memorizzate all'interno della macchina stessa ed eseguite automaticamente, senza interventi esterni. È qui che compare, per la prima volta in maniera esplicita e “organica” (in un senso cibernetico), il *principio del programma memorizzato*. Fa propria, inoltre, la lezione di Turing

⁴⁷¹ As a consequence, the solution of wide classes of non-linear partial differential equations will be easily managed in the EDVAC, whereas such problems can be carried out on the ENIAC only with the aid of a much slower external memory capacity in the form of punched cards.

⁴⁷² “An automatic computing system is a (usually highly composite) device which can carry out instructions to perform calculations of a considerable order of complexity - e.g. the numerical solution of a partial differential equation in 2 or 3 independent variables”.

sulla macchina universale, ovvero l'equivalenza numerica di ciò che è dato come istruzione e di ciò che è trattato come dato. In questo nuovo paradigma, "una volta che queste istruzioni sono state fornite al dispositivo, esso deve essere in grado di eseguirle completamente e senza alcun bisogno di ulteriore intervento intelligente umano [...] I risultati sono dati numerici" [von Neumann, 1945, 1.2, trad. mia]⁴⁷³.

Nelle macchine analizzate finora, la programmabilità instaurava una relazione *simbiotica*, finalizzata alla risoluzione di un compito (crittoanalitico o computazionale), tra la struttura e i vincoli fisici della macchina da un lato, e l'intelligenza e la capacità operativa umana dall'altro, in particolare quella dell'operatrice, che fungeva al contempo da *interfaccia* e da *sistema operativo*. Ora, invece, il "controllo intelligente" - per quanto l'espressione possa apparire problematica - si trasferisce dalla mente e dalle mani delle WREN o delle programmatrici dell'ENIAC all'interno della macchina stessa, dove era sinora solo potenziale come insieme di cavi e valvole. Il *computer* umano che popolava Bletchley Park, Aberdeen o il campus dell'Università della Pennsylvania diventa un *computer* meccanico, digitale ed elettronico, grazie all'incorporazione nel calcolatore delle istruzioni.

In *First Draft* "il flusso di istruzioni e dati rispecchiava il modo in cui gli esseri umani svolgevano calcoli scientifici, come una serie di operazioni matematiche, utilizzando calcolatrici meccaniche, tavole numeriche e carta e penna" [Haigh, Ceruzzi, 2021, p. 16 trad. mia]⁴⁷⁴. Il programma è scritto come una serie di istruzioni in codice, gli "*orders*" [von Neumann, 1945, 14.1, 14.2; 15.3], e queste vengono memorizzate assieme ai dati da elaborare. Dati e istruzioni vengono lette dagli stessi meccanismi. Le istruzioni erano accompagnate da parametri, gli indirizzi, per la loro allocazione in memoria.

Ricorda Burks a questo proposito: "Pres [Presper Eckert] e John [Mauchly] inventarono la memoria a linea di ritardo con mercurio in ricircolo, con una capacità sufficiente a contenere sia le istruzioni del programma che i dati. von Neumann ideò il primo codice operativo moderno e sviluppò la progettazione logica di un calcolatore elettronico in grado di eseguirlo" [Burks, 1980, p. 312, trad. mia]⁴⁷⁵. Combinate, le due innovazioni rendevano possibile la costruzione di

⁴⁷³ "Once these instructions are given to the device, it must be able to carry them out completely and without any need for further intelligent human intervention [...]. The results are numerical data".

⁴⁷⁴ "The flow of instructions and data mirrored the way humans performed scientific calculations as a series of mathematical operations, using mechanical calculators, books of tables, and pencil and paper".

⁴⁷⁵ "Pres and John invented the circulating mercury delay line store, with enough capacity to store program information as well as data. Von Neumann created the first modern order code and worked out the logical design of an electronic computer to execute it".

un calcolatore *general-purpose*, in cui dati e istruzioni erano memorizzati all'interno della macchina stessa

Nel *First Draft* sono previsti “8 tipi di ordine” che impiegano 3 cifre per distinguersi l'uno dall'altro [von Neumann, 1945, 15.4]⁴⁷⁶. Quattro di questi ordini impiegavano altri 4 bit usati come parametri per scegliere tra un gruppo di dieci operazioni logiche e aritmetiche. E diverse istruzioni avevano indirizzi a 13 bit. Ogni *order* variava tra i 9 e i 22 bit [Cfr. von Neumann, 1945, 15.4, 15.5; Haigh et. al, 2016, p.147-148.].

Un numero così limitato di istruzioni, corredate da indirizzi e parametri, implica che sia affidato alla componente hardware – in particolare ai circuiti di controllo logico – il compito di decodificare ogni parte dell'istruzione, selezionare l'operazione da eseguire e gestire gli indirizzi. Questo approccio si contrappone alla filosofia proposta da Turing per l'ACE nel *Proposed Electronic Calculator*, che abbiamo già incontrato, fondata sull'idea di mantenere l'hardware più snello possibile e di lasciare al software la gestione della complessità [Cfr. Haigh et al., 2016, p.148].

L'esecuzione delle operazioni seguiva un ciclo ben definito: prelievo dell'istruzione dalla memoria (in termini contemporanei, *fetch*), interpretazione del suo contenuto ed esecuzione dell'operazione indicata. Questo ciclo seriale era regolato da un meccanismo di controllo centrale, responsabile della corretta sequenza delle operazioni. L'indirizzo dell'istruzione successiva era conservato in una posizione dedicata della memoria e incrementato automaticamente al termine di ogni ciclo, consentendo così l'elaborazione sequenziale del programma. La possibilità, offerta da specifiche istruzioni, di modificare tale indirizzo rendeva possibile deviare il normale ordine di esecuzione, permettendo ramificazioni condizionate e ritorni da subroutine, secondo quanto previsto dal codice operativo [cfr. Haigh, Ceruzzi, 2021, pp. 16-17; Haigh et al., 2016, p. 148].

I limiti dell'ENIAC in termini di generalità venivano parzialmente compensati attraverso un approccio modulare e l'adozione di sistemi *multiprocessing* e di elaborazione parallela [Cfr. Eckert, 1980, p.528]. L'EDVAC una macchina interamente *seriale*.

Burks fornisce una felice sintesi delle differenze tra i due calcolatori: “Poiché l'EDVAC sarebbe stato molto più veloce, compatto e semplice rispetto all'ENIAC, non vi era più la necessità di ricorrere al parallelismo per ottenere velocità, e si decise quindi di memorizzare e

⁴⁷⁶ “To begin with we have in $(\alpha) - (\eta)$ 8 types of order, to distinguish these from each other requires 3 digits”.

processare i numeri in modo seriale. Il principio guida della progettazione dell'EDVAC era: una cosa alla volta, fino all'ultimo bit!" [Burks, 1980, p.338 trad. mia]⁴⁷⁷.

Le considerazioni appena fatte ci indirizzano verso il terzo punto di svolta del "paradigma" EDVAC: quello relativo all'*architettura*. Il calcolatore delineato in *First Draft* è composto dalle parti seguenti: una che presiede le operazioni aritmetiche del calcolatore, la *Central Arithmetic* (CA); l'unità di controllo logico centrale del calcolatore, il *Central Control* (CC); la memoria totale del dispositivo (M); i sistemi di input (I) e output (O) e un supporto esterno di registrazione del dispositivo, destinato alla lettura e scrittura delle informazioni tra la macchina e l'esterno, (R). [von Neumann, 1945, 2.2-2.8].

Prima di considerare la disposizione architetturelle delle componenti del calcolatore descritto nel rapporto, è opportuno esaminare la cornice teorica entro cui esse vengono trattate. Nella discussione di queste parti della macchina e del sistema che esse nel complesso costituiscono, von Neumann adotta la metafora cibernetica dell'analogia tra componenti artificiali e apparato neurofisiologico e senso-motorio naturale dell'essere umano. "Può sembrare anomalo descrivere il progetto di una tecnologia in termini biologici; così appariva certamente a Eckert e Mauchly. Ma questa pratica permise a von Neumann di separare la progettazione logica dai componenti ingegneristici utilizzati per realizzarne la struttura logica" [Aspray, 1990, p. 173, trad. mia]⁴⁷⁸.

Nel progetto dell'EDVAC culmina e si diffonde quella traiettoria di identificazione tra computazione, elaborazione dell'informazione, logica, hardware artificiale e *wetware* che ha attraversato il discorso fin qui sviluppato. Grazie all'influenza del rapporto e del suo ideatore, infatti, questa macchina diventerà il paradigma concreto che ispirerà la realizzazione pratica dei calcolatori successivi.

Gli interessi di von Neumann nella teoria dell'informazione risalgono già alla fine degli anni Venti, con i suoi studi sulle fondazioni matematiche della meccanica quantistica. Uno dei paradossi che angustiarono i teorici della nuova fisica era quello del demone di Maxwell: un essere ipotetico capace di osservare il moto delle singole molecole di un gas e di separarle attraverso una valvola, lasciando passare solo le molecole più veloci in una direzione e le più

⁴⁷⁷ "Because EDVAC would be so much faster, smaller, and simpler than ENIAC, there was no longer the need for parallelism to gain speed, and it was decided to store numbers serially and process them serially. The guiding principle of EDVAC design was: One thing at a time, down to the last bit!"

⁴⁷⁸ "It may seem anomalous to capture the design of a technology in biological terms; it certainly did seem that way to Eckert and Mauchly. But this practice allowed von Neumann to separate the logical design from the engineered components used to implement the logical structure".

lente nell'altra, producendo così una differenza di temperatura senza compiere lavoro - in apparente violazione del secondo principio della termodinamica. [Cfr. Maxwell, 1872 in Maxwell, 1970, pp.308-309]⁴⁷⁹

Il paradosso fu risolto nel 1929 da Leó Szilárd (1898-1964), un altro dei Marziani ungheresi che lavorarono al progetto Manhattan, il quale dimostrò che la chiave risiedeva nella relazione tra entropia e informazione: per acquisire informazioni sullo stato delle particelle, il demone deve introdurre nel sistema una quantità equivalente di entropia negativa, che compensa esattamente la riduzione di entropia prodotta dalla separazione. In *Mathematische Grundlagen der Quantenmechanik*, von Neumann fece propria l'idea di Szilárd ipotizzando l'intervento di un essere intelligente all'interno di un sistema termodinamico [Cfr. Aspray, 1990, p.173-174, von Neumann, cap.V, 1932].

In modo parallelo a quanto avveniva con Wiener - che riconosceva in Maxwell un precursore della cibernetica [Cfr. Wiener, 1948, pp.34-35] - e con le macchine di Bletchley, che traducevano *in hardware* raffinate procedure probabilistiche per decifrare messaggi crittati in modo pseudorandomizzato, anche la strada che ha condotto al calcolatore immaginato da von Neumann può essere fatta risalire a quella svolta statistica che avrebbe portato alla teoria dell'informazione. Vedremo in seguito come il rapporto tra computazione e informazione studiato da von Neumann nella sua teoria degli "automi" contribuirà al mutamento del design del calcolatore che porta il suo nome.

Il matematico ungherese era entrato in contatto, all'inizio degli anni Quaranta, con Shannon⁴⁸⁰ e con il gruppo formato da Wiener, Arturo Rosenblueth e Julian Bigelow, che aveva da poco pubblicato il seminale articolo su comportamento, scopo e teleologia [Rosenblueth et

⁴⁷⁹ "But if we conceive a being whose faculties are so sharpened that he can follow every molecule in its course, such a being, whose attributes are still as essentially finite as our own, would be able to do what is at present impossible to us. For we have seen that the molecules in a vessel full of air at uniform temperature are moving with velocities by no means uniform, though the mean velocity of any great number of them, arbitrarily selected, is almost exactly uniform. Now let us suppose that such a vessel is divided into two portions, A and B, by a division in which there is a small hole, and that a being, who can see the individual molecules, opens and closes this hole, so as to allow only the swifter molecules to pass from A to B, and only the slower ones to pass from B to A. He will thus, without expenditure of work, raise the temperature of B and lower that of A, in contradiction to the second law of thermodynamics".

⁴⁸⁰ Shannon aveva conosciuto von Neumann nel suo anno di post-doc allo IAS dal 1940 al 1941. Nel testo introduttivo a *The Mathematical Theory of Communication* [Shannon, 1948], Warren Weaver ricorda come fu von Neumann a notare il legame tra il lavoro di Claude Shannon e la fisica statistica di Boltzmann (1894), che definiva l'entropia come "informazione mancante", cioè come misura del numero di alternative ancora possibili per un sistema fisico una volta registrate tutte le informazioni osservabili. Ulteriore profondità storica alla trattazione matematica dell'informazione come concetto fisico è individuata, secondo Weaver, negli studi di Szilárd e di von Neumann, riconosciuti tra le basi teoriche del lavoro di Shannon, insieme ad antecedenti più immediati come Nyquist e Hartley [Weaver, 1948, pp.1-2 n.1 trad. it].

al., 1948], in cui venivano annullate le differenze funzionali tra sistemi artificiali e organici, entrambi concepiti come sistemi a retroazione negativa. Furono proprio questi ultimi⁴⁸¹ a spingere von Neumann alla lettura di *A Logical Calculus of the Ideas Immanent in Nervous Activity* di McCulloch e Pitts [McCulloch, Pitts, 1943] in cui – come abbiamo visto – veniva presentato un modello idealizzato di neurone, equivalente a un automa computazionale a stati finiti.

Ispirato dal lavoro di McCulloch e Pitts, citati esplicitamente per ben due volte nel *First Draft* [von Neumann, 1945, 4.2, 6.4], von Neumann fa propria l'analogia formale tra biologia e computazione, tra attività neuronale ed elaborazione – discreta, binaria e digitale – dell'informazione. La metafora cibernetica inizia, almeno su carta, nel *First Draft*, a materializzarsi in un calcolatore.

Le varie parti del calcolatore sono presentate e trattate come *organi* [*organs*]⁴⁸² del sistema EDVAC e tra queste, “le tre parti specifiche CA, CC (insieme C) e M corrispondono ai neuroni associativi nel sistema nervoso umano”, mentre gli organi di input-output sono descritti da von Neumann, impadronitosi del vocabolario della cibernetica, come “neuroni sensoriali o afferenti” e “motori o efferenti” [von Neumann, 1945, 2.6 trad. mia]⁴⁸³.

Scriva von Neumann in *First Draft*, “I neuroni degli animali superiori [...] hanno un comportamento “tutto-o-nulla”, cioè due stati: quiescente e eccitato”. Si comportano come quegli elementi presenti in ogni dispositivo calcolatore digitale “contiene certi elementi simili a relè, con equilibri discreti. Un tale elemento possiede due o più stati distinti nei quali può esistere indefinitamente”. Un elemento del genere può essere implementato da “ruote” [*wheels*]; “relè telegrafici” [*telegraph relays*] elettromeccanici o “tubi a vuoto” [*vacuum tubes*]

⁴⁸¹ Peter Galison mette in evidenza l'estremo interesse di Norbert Wiener nel riunire un gruppo di esperti provenienti da campi scientifici apparentemente eterogenei – ingegneria della comunicazione, ingegneria dei calcolatori, neurofisiologia, statistica – sottolineando come “che le relazioni tra questi campi di ricerca hanno sviluppato un grado di vicinanza che rende un incontro informale [...] altamente desiderabile.” [Galison, 1994, p.62 trad. it.] A questo gruppo *in fieri*, Wiener, John von Neumann e Howard H. Aiken diedero il nome di “*Teleological Society*”. Il primo incontro della Società Teleologica si tenne il 6-7 gennaio 1945. All'incontro parteciparono, tra gli altri, anche McCulloch, Pitts e il neurofisiologo spagnolo Rafael Lorente de Nó (1902-1990), pioniere della ricerca sui neuroni, visti come circuiti ricorrenti e interconnessi alla base delle funzioni cognitive e motorie. Wiener fu entusiasta dell'esito dell'incontro “Alla fine”, Wiener disse con impeto a Rosenblueth, “fummo tutti convinti che la disciplina che comprende sia gli aspetti ingegneristici che neurologici è essenzialmente la stessa” [Galison, 1994, p.62 trad. it.].

⁴⁸² Anche Turing in *Proposed Electronic Calculator*, si approprierà della descrizione delle parti dell'ACE come organi e, come abbiamo notato, seguirà, per via indiretta, i diagrammi logici di McCulloch-Pitts presenti in *First Draft*.

⁴⁸³ “The three specific parts CA, CC (together C) and M correspond to the associative neurons in human nervous system. it remains to discuss the equivalents of the sensory or afferent and the motor or efferent neurons, These are the input and the output organs of the device”.

elettronici. Seguendo McCulloch e Pitts e la loro modellizzazione del neurone, von Neumann può dichiarare “È facile vedere che queste funzioni neuronali semplificate possono essere imitate da relè telegrafici o da tubi a vuoto” [von Neumann, 1945, 4.1-4.3 trad. mia]⁴⁸⁴.

Nelle pagine di un oscuro rapporto tecnico prendeva forma quel “cervello meccanico” favoleggiato dalla stampa dell’epoca e da testi come *Giant Brain or Machines that Think* del 1949 [Berkeley, 1949], ad opera del computer scientist, divulgatore e cofondatore nel 1947 della *Association for Computing Machines* (ACM), Edmund C. Berkeley (1909-1988). Il volume si apriva con queste parole:

L’argomento di questo libro è un tipo di macchina che si avvicina più di qualsiasi altra, prima del 1940, a essere un cervello che pensa. Queste nuove macchine vengono talvolta chiamate cervelli meccanici, talvolta calcolatori a controllo sequenziale, e altre volte con altri nomi. In sostanza, però, si tratta di macchine capaci di manipolare informazioni con grande abilità e grande velocità. E questo potere è molto simile al potere di un cervello [Berkeley, 1948, p.vii trad. mia]⁴⁸⁵.

Nel quadro dell’impiego di un vocabolario biologista di matrice cibernetica, gli organi della macchina - C, M, R, I, O - nella svolta architettureale proposta da von Neumann e nel design logico e implementativo dell’EDVAC delineato nel *First Draft* erano organizzati in modo rigorosamente seriale. Le operazioni non erano distribuite in parallelo, come nell’ENIAC, né eseguite immediatamente una volta lette dal nastro, come accadeva nelle macchine elettromeccaniche. Il programma, memorizzato, seguiva una sequenza ben definita di passaggi: veniva caricato nella macchina dal medium R, che conteneva l’informazione, e trasferito in M, la memoria centrale, collegata linearmente con l’unità di controllo.

La serializzazione dell’elaborazione dell’informazione era integrale: una sola operazione veniva eseguita alla volta, e tutti i trasferimenti tra memoria e unità aritmetico-logica passavano attraverso un canale unico. Il controllo centrale orchestrava l’intero processo, gestendo il flusso delle istruzioni attraverso registri specializzati coordinando in modo

⁴⁸⁴ “It is worth mentioning, that the neurons of the higher animals are definitely elements in the above sense. They have all-or-none character that is two states: Quiescent and excited.”; “Every digital computing device contains certain relay like elements, with discrete equilibria. Such be an element has two or more distinct states in which it can exist indefinitely”; “It is easily seen that these simplified neuron functions can be imitated by telegraph relays or by vacuum tubes”.

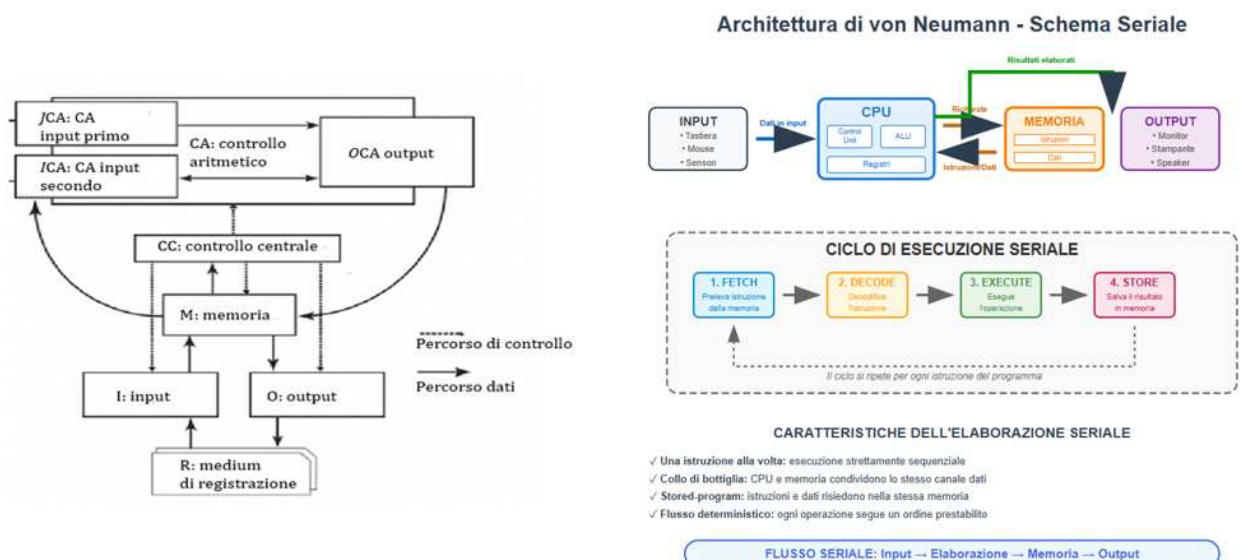
⁴⁸⁵ The subject of this book is a type of machine that comes closer to being a brain that thinks than any machine ever did before 1940. These new machines are called sometimes mechanical brains and sometimes sequence-controlled calculators and sometimes by other names. Essentially, though, they are machines that can handle information with great skill and great speed. And that power is very similar to the power of a brain.

sequenziale le fasi del calcolo [Cfr. Ceruzzi, Haigh, 2021, p.6; pp.15-16; Haigh et al., p. 2016, pp.144-146.]

Questa è l'architettura di von Neumann [fig. 21], che ancora oggi costituisce la base dei computer *general purpose*, "composto da unità per l'aritmetica, la memoria, il controllo, l'input e l'output". Dati, risultati intermedi e istruzioni sono memorizzati internamente nella stessa memoria, in forma numerica codificata [Aspray, 1990, p. 64, trad. mia], fondata su una logica in sequenza seriale del trattamento dell'informazione.

L'architettura di questo computer, molto più snella e veloce rispetto alle macchine precedenti, rappresenterà non solo un modello ingegneristico, ma anche, come abbiamo visto, un paradigma fondamentale per le scienze cognitive e l'Intelligenza Artificiale simbolica. A livello di implementazione, essa costituirà l'analogo concretamente realizzato della macchina universale di Turing: un modello della mente umana intesa come sistema computazionale di elaborazione dell'informazione. [Cfr. Frixione, 1992, pp. 13-14].

Fig.21 L'architettura di von Neumann



Ricorda Goldstine come prima dell'entrata in scena di von Neumann e della sua *expertise* intorno ai problemi di logica e matematica fondamentale – e della conseguente leadership

naturale che assunse riguardo a questi problemi – il gruppo a lavoro su ENIAC si fosse concentrato principalmente su problemi tecnologici. Con von Neumann in gioco, sebbene questo rappresentasse una “perfetta divisione del lavoro”, “il gruppo finì così per dividersi in due: da una parte i tecnologi - Eckert e Mauchly - e dall'altra i logici - von Neumann, Burks e io” [Goldstine, 1972, p.188 trad. mia]⁴⁸⁶.

La polarizzazione del team - già appesantita dalle crescenti tensioni relative alla questione dei brevetti tra Mauchly ed Eckert e l'amministrazione dell'Università della Pennsylvania, cui era affiliata la Moore School - fu aggravata dal fatto che i due cominciarono a percepire come proprie idee, rimaste coperte dal segreto militare, venissero sempre più spesso attribuite esclusivamente a von Neumann. Tra queste figuravano, ad esempio, quelle contenute nel rapporto *Disclosure of a Magnetic Calculation Machine*, redatto da Presper Eckert il 29 gennaio 1944 [Eckert, 1944] e rimasto secretato per decenni. Invece, von Neumann poteva discutere liberamente di problemi di logica del calcolatore, anche grazie al vocabolario cibernetico che impiegava nella trattazione.

Il gruppo si divise ed EDVAC rimase orfano prima di nascere⁴⁸⁷. Gli eventi culminarono in una lunga controversia legale, avviata dopo che, il 26 giugno 1947, Mauchly ed Eckert tentarono di brevettare l'ENIAC. La causa si protrasse per oltre venticinque anni, fino al 1973, quando il brevetto fu definitivamente invalidato [Cfr. Haigh et al., 2016, p.108; p.267]⁴⁸⁸. Nel frattempo, i due avevano abbandonato la Moore School e, nel 1946, fondato la *Electronic Control Company*, che l'anno successivo divenne *Eckert-Mauchly Computer Corporation* (EMCC) e che, nel 1950, sarebbe stata acquisita dalla Remington Rand - una sorta di antesignana delle odierne start-up informatiche, alla quale si collegheranno più avanti sviluppi di rilievo.

Anche von Neumann abbandonò la Moore School. Tornò all'inizio del 1946 allo IAS per costruire, secondo le sue idee, un calcolatore “concepito come strumento scientifico” [Dyson,

⁴⁸⁶ “Prior to his appearance on the scene, the group at the Moore School concentrated primarily the technological problems, which were very great: after his arrival he took over leadership on the logical problems. Thus the group tended to split into the technologists - Eckert and Mauchly - and the logicians in von Neumann, Burks, and I”.

⁴⁸⁷ I lavori di costruzione di EDVAC iniziarono nel 1949 all'Aberden Proving Ground. Il calcolatore divenne operativo il 1951 e iniziò a funzionare produttivamente il 1952. Per nove mesi del 1952, l'EDVAC fu disponibile in media per 47,4 ore settimanali, suddivise tra 23,3 ore di verifica del codice e 24,1 ore di lavoro produttivo. Parallelamente, il tempo medio settimanale dedicato agli interventi ingegneristici ammontava a 104,8 ore, delle quali ben 70,4 erano assorbite da operazioni di manutenzione non programmata. In quel periodo vennero sostituiti circa 10.000 tubi difettosi più del doppio dei 3000 installati. Huskey, 2003b. Per approfondire le dinamiche implementative di EDVAC si veda Williams, 1993.

⁴⁸⁸ Su questa lunghissima controversia legale e sull'impatto che essa ha avuto sulla ricostruzione storica degli eventi intorno alla costruzione dell'ENIAC e al ruolo dei diversi protagonisti si rimanda a Haigh, et al., 2016, pp.265-269.

2012, p.94 trad. it.]⁴⁸⁹. Portò con sé Goldstine e Burks. Al gruppo si unì Julian Bigelow, che aveva lavorato con Wiener al predittore antiaereo, per occuparsi della parte ingegneristica.

Il progetto del calcolatore - che riprendeva l'architettura, i principi e il trattamento delle componenti come organi delineati nel rapporto sull'EDVAC - trovò la sua prima formalizzazione in *Preliminary Discussion of the Logical Design of an Electronic Computing Instrument* [Burks et al., 1946]. Questo documento, che al pari del *First Draft* sull'EDVAC mise a dura prova i ciclostili impiegati per la riproduzione delle copie, fu redatto nell'ufficio adiacente a quello di Gödel, uno degli ispiratori ideali della logica che la macchina mirava a implementare [Dyson, 2012, p. 119, trad. it.].

In esso, i tre autori delineano un "calcolatore a uso generale", completamente automatico, dotato di quattro organi principali: aritmetico, di memoria, di controllo e di connessione con l'operatore umano [Burks et al., 1946, 1.1 trad. mia]⁴⁹⁰.

Il calcolatore proposto rappresentava, in coerenza con i principi di design logico analizzati, la realizzazione fisica di una "macchina universale" [all-purpose], capace cioè di essere istruita affinché esegua qualsiasi calcolo esprimibile in termini numerici [Burks et al., 1946, 1.2, trad. mia]⁴⁹¹. Perché tale universalità fosse garantita, era necessario che la macchina non memorizzasse soltanto i numeri, ma anche il programma: al pari dell'EDVAC, essa doveva dunque implementare il principio dello *stored program*. La macchina, digitale e basata sull'aritmetica binaria [Burks et al., 1946, 5.2], doveva pertanto includere un'unità di controllo capace di eseguire automaticamente le istruzioni memorizzate, senza ulteriori interventi umani [Burks et al., 1946, 1.4]⁴⁹².

⁴⁸⁹ Per un approfondimento sul calcolatore dello IAS, sul suo design logico (memoria unica per dati e istruzioni, architettura parallela, controllo elaborato), sulle specifiche tecniche e sulla storia della sua effettiva realizzazione, completata nel 1951 e formalmente inaugurata nel giugno 1952, si rimanda ad Aspray 1990, pp.59-89. Per le idee di von Neumann sul calcolatore dello IAS come strumento scientifico, concepito per essere impiegato in un'ampia gamma di campi di ricerca - dalle equazioni differenziali ordinarie e parziali all'algebra lineare numerica, dalla matematica pura e statistica all'astrofisica, fluidodinamica, fisica atomica e nucleare, fino a cristallografia, chimica, storia della scienza, simulazioni di traffico ed esperimenti di "bionumeric evolution" - e per la documentazione delle principali applicazioni sviluppate nei primi anni di attività della macchina, si veda Aspray, 1990, pp. 155-171.

⁴⁹⁰ "Inasmuch as the completed device will be a general-purpose computing machine it should contain certain main organs relating to arithmetic, memory-storage, control and connection with the human operator. It is intended that the machine be fully automatic in character, i.e. independent of the human operator after the computation starts".

⁴⁹¹ "Or an all-purpose machine it must be possible to instruct the device to carry out any computation that can be formulated in numerical terms".

⁴⁹² "If the memory for orders is merely a storage organ there must exist an organ which can automatically execute the orders stored in the memory. We shall call this organ the *Control*".

Nel progetto originario, la memoria e il controllo avrebbero dovuto basarsi su circa 2.300 valvole termoioniche *Selectron*, sopra analizzate, sviluppate dalla RCA. Tuttavia, nella realizzazione effettiva, queste furono sostituite dai tubi Williams e da valvole triodo 6J6, più economiche e affidabili, usate come *flip-flop*, introdotte da Julian Bigelow [Burks et al., 1946, 4.1 e 6.2; Dyson, 2012, pp. 144–146, trad. it.]⁴⁹³.

Per quanto riguarda la “connessione con l’essere umano”, *Preliminary Discussion* progettava l’esistenza di organi “attraverso i quali l’operatore umano e la macchina possano comunicare tra loro” [Burks et al., 1946, 1.6 trad. mia]⁴⁹⁴. Questi organi corrispondono ai sistemi di input-output del dispositivo, i quali costituiscono una forma secondaria di memoria automatica del calcolatore e permettono all’essere umano di inserire *words* nella macchina e di leggerne i risultati. Nel documento, oltre ai sistemi che registrano l’informazione su supporti fisici come nastri o fili [*wire*], si prevedeva anche la possibilità di una visualizzazione grafica dei risultati delle computazioni, resa possibile dalla fluorescenza del raggio di elettroni delle valvole *Selectron* che si intendeva adottare [cfr. Burks et al., 1946, 4.6–4.8].

La serialità dell’elaborazione dell’informazione nel calcolatore a programma memorizzato, e la sequenzialità del traffico su un unico canale [*bus*] di dati tra le unità di controllo e quelle di memoria, insieme allo sbilanciamento tra la velocità del processore e quella della memoria, produce un fenomeno noto come collo di *bottiglia di von Neumann* [*von Neumann Bottleneck*]. [Cfr. Zou et al., 2021; Edwards, O’Keefe, 2016; Dyson, 2012, p.92 trad. it.].

Questo termine – coniato dall’informatico e matematico statunitense John Backus (1924–2007) nella sua *lectio* di accettazione del Premio Turing del 1977 [Backus, 1978] – descrive un limite architetturale fondamentale del “paradigma EDVAC” che affligge anche i computer contemporanei (in cui la CPU, *Central Processing Unit*, accorpa le originarie unità di controllo e aritmetico-logica): le unità di controllo, molto più veloci, sono costrette ad attendere il trasferimento dei dati da e verso la memoria, rallentando drasticamente l’elaborazione effettiva, in attesa o va sbattere contro un muro, il “*memory wall*” [Zou et al., 2021].

Backus, nella sua *Turing Award Lecture*, osserva come un calcolatore di tipo von Neumann, ridotto alla sua forma più semplice, consista in una “unità di elaborazione centrale (o CPU), una memoria e un condotto di collegamento che può trasmettere una sola parola alla volta tra la CPU e la memoria. Propongo di chiamare questo tubo il *collo di bottiglia di von*

⁴⁹³ L’analisi tecnica del lavoro di Bigelow sulla macchina dello IAS è descritta dallo stesso ingegnere in Bigelow, 1980, pp.291-310.

⁴⁹⁴ “Lastly there must exist devices, the input and output organ, whereby the human operator and the machine can communicate with each other”.

Neumann". Il traffico che attraversa tale collo di bottiglia, ironicamente, "non consiste in dati utili, ma semplicemente in nomi di dati, oltre che in operazioni e dati impiegati unicamente per calcolare tali nomi".

Ciò che conta maggiormente, tuttavia, è che il *von Neumann bottleneck* non va inteso soltanto come un vincolo implementativo, fisico o di design. A riprova del peso e dell'influenza esercitati dall'architettura ideata dal matematico ungherese sull'intera *computer science*, esso rappresenta soprattutto un vero e proprio "collo di bottiglia intellettuale che ci ha mantenuti vincolati a un modo di pensare basato sull'elaborazione di una parola alla volta, invece di incoraggiarci a ragionare in termini di unità concettuali più ampie, legate al compito da svolgere". In questa prospettiva, la programmazione si è ridotta a "pianificare e dettagliare l'enorme traffico di parole attraverso il collo di bottiglia di von Neumann", un traffico generato non tanto dai dati significativi in sé, quanto dai modi per indicare dove essi si trovino in memoria [Backus, 1978, trad. mia]⁴⁹⁵.

George Dyson afferma come von Neumann fosse consapevole di questa difficoltà strutturale del suo design. Riporta infatti uno stralcio di una lettera scritta il 19 marzo 1946 a Max Newman – che, dopo la visita a Princeton, importerà a Manchester i criteri del calcolatore in costruzione allo IAS – in cui si afferma che "il sistema nel suo complesso sarà ben equilibrato e dunque, se usato in modo adeguato e intelligente, non ci sarà alcun collo di bottiglia", e questo anche in connessione con l'operatore umano. Non ci sarà, scrive, nessun *bottleneck* "nemmeno negli output e input delle menti umane cui dovrà essere abbinato". [von Neumann, lettera a M. Newman, 19 marzo 1946, in Dyson, 2012, p. 92, trad. it.].

Tuttavia, è necessario approfondire la questione sollevando alcuni interrogativi: in che modo il collo di bottiglia che affligge l'architettura del calcolatore di von Neumann - ostacolo non solo implementativo ma anche epistemologico - si rapporta all'epistemologia materiale della mente e del cervello umano, inteso come elaboratore di informazione, quando interagisce con una macchina costruita secondo tale architettura? È legittimo adottare la metafora biologista della cibernetica - come fece von Neumann, pur con le cautele metodologiche del caso

⁴⁹⁵ In its simplest form a von Neumann computer has three parts: a central processing unit (or CPU), a store, and a connecting tube that can transmit a single word between the CPU and the store (and send an address to the store). I propose to call this tube the von Neumann bottleneck. [...] Ironically, a large part of the traffic in the bottleneck is not useful data but merely names of data, as well as operations and data used only to compute such names [...] Not only is this tube a literal bottleneck for the data traffic of a problem, but, more importantly, it is an intellectual bottleneck that has kept us tied to word- at-a-time thinking instead of encouraging us to think in terms of the larger conceptual units of the task at hand. Thus programming is basically planning and detailing the enormous traffic of words through the von Neumann bottleneck, and much of that traffic concerns not significant data itself but where to find it.

- e trattare le parti del calcolatore come organi, secondo i principi di design presenti nel *First Draft* e in *Preliminary Discussion*? Oppure occorre riformularne la logica per garantirne una trattazione logico-matematica rigorosa, come quella proposta da Turing?

Proprio in riferimento al lavoro del matematico inglese sulle macchine non organizzate, concepite per superare i limiti intrinseci delle macchine universali, ci si può allora domandare: una macchina di Turing universale realizzata fisicamente secondo l'architettura von Neumann è sufficiente a coprire l'intera gamma delle facoltà del cervello umano? È un modello adeguato della mente? È possibile - o sufficiente - spiegare il cervello e la sua attività secondo i criteri di serialità, natura digitale, generalità computazionale e complessità?

Analizzeremo a questo proposito il punto di vista dello stesso von Neumann, esemplificato da una lunga lettera indirizzata a Wiener, in cui venivano messi in luce i limiti di un approccio cibernetico fondato sulla logica formale del modello di McCulloch e Pitts e si prefigurava la necessità di uno studio sugli automi. In questa direzione si intravedono i prodromi concettuali di un tipo di computazione - alla base dell'attuale configurazione dell'intelligenza artificiale - diverso dall'architettura di von Neumann e più vicino alla morfologia del sistema nervoso, fondato sull'analisi delle componenti "minimali", biologiche e artificiali, dei sistemi di elaborazione dell'informazione. L'integrazione della teoria dell'informazione di Shannon nel cuore della logica degli automi contribuì a dare a tale prospettiva una forma stocastico-probabilistica.

Proveremo più avanti a delineare una risposta a queste domande, approfondendo la riflessione di von Neumann sul rapporto tra cervello e calcolatori e precisando il paradigma alternativo, più volte richiamato, rispetto a quello incarnato dall'architettura seriale del calcolatore: il connessionismo, che ci condurrà alle soglie dell'attuale configurazione dell'elaborazione digitale dell'informazione e dell'intelligenza artificiale contemporanea.

Prima di affrontare questa questione è necessario prendere in esame alcune conseguenze dell'intricatissima genealogia della *lignée* ENIAC-EDVAC⁴⁹⁶. Nelle pagine che seguono prenderemo in esame la diffusione, la "diaspora", del "paradigma EDVAC", che non solo ha orientato gli sviluppi della computazione scientifica e commerciale, ma è divenuto anche -

⁴⁹⁶ A dare il senso alla miriade di ramificazioni dell'eredità ENIAC-EDVAC è l'insieme di macchine i cui nomi non sfuggirebbero in un romanzo di Asimov. Negli Stati Uniti, il modello ispirò realizzazioni come SEAC, BINAC, UNIVAC, AVIDAC, IBM 701, ILLIAC, JOHNNIAC, MANIAC, ORACLE, ORDVAC, SWAC e WHIRLWIND. Al di fuori del contesto americano, la stessa eredità prese forma in Europa e oltre, con realizzazioni come l'EDSAC e il Manchester Mark I nel Regno Unito, il BESK in Svezia, il DASK in Danimarca, lo SILLIAC in Australia, lo SMIL ancora in Svezia e il WEIZAC in Israele [Cfr. Burks, 1980 p. 336; Bruderer, 2020 p. 1009].

con conseguenze tutt'altro che neutrali dal punto di vista socioculturale - un criterio di riorganizzazione della società. Parallelamente, approfondiremo lo sviluppo tecnologico dell'elettronica a stato solido: dai relè e dai tubi catodici, di cui abbiamo già incontrato le prime applicazioni, per poi seguirne gli ulteriori sviluppi con l'introduzione del transistor, dei circuiti integrati e infine del microprocessore: un percorso che ha determinato profonde metamorfosi del concetto stesso di calcolatore - dai giganteschi "cervelli elettronici" dei primi anni alla stagione dei minicomputer, del time-sharing e infine del personal computer - e ha ridefinito di volta in volta le configurazioni del rapporto tra esseri umani e macchine. Sistemi operativi, programmi e nuove interfacce hanno progressivamente offerto risposte diverse all'interrogativo fondamentale della relazione tra sistemi naturali e artificiali di elaborazione dell'informazione.

Lo sviluppo tecnologico in hardware ha dato concretezza alla dinamica culturale e contro culturale che, dalla distopia cibernetica, ha condotto verso la computazione personalizzata. In tale prospettiva, i passaggi attraverso transistor, circuito integrato e microprocessore possono essere letti come ulteriori approfondimenti della metafora cibernetica di analogia funzionale tra il calcolatore e le sue componenti da un lato, e il cervello umano dall'altro. La stessa metafora ha trovato nuove traduzioni sociologiche, fino a spiegare perché Norbert Wiener ha potuto diventare, come ricordato all'inizio di questo lavoro, una figura citata perfino nelle riviste della contestazione.

Il viaggio che ci ha condotto attraverso i vari stadi della computazione - dal Congresso Internazionale dei Matematici di Parigi del 1900, con i Problemi di Hilbert, passando per l'Inghilterra (Bletchley Park, Londra, Manchester), e tornando quindi in America a Princeton, ad Aberdeen e alla Moore School di Philadelphia, e che ci porterà fino al cuore della Silicon Valley - richiede tuttavia di aprire un ulteriore spazio ed esplorarlo: quello digitale, l'universo dei dati e della simulazione, che il lavoro sui computer in tempo di guerra ha contribuito a dischiudere.

Cap.5: Diffusioni digitali: serialità, elettronica, società

1. *Rien ne va plus: la nascita della simulazione*

Nel 1946, come abbiamo visto, il gruppo della Moore School si divise. John Mauchly e J. Presper Eckert lasciarono l'Università della Pennsylvania per fondare una propria azienda di calcolatori, mentre Arthur Burks e Herman Goldstine si unirono a John von Neumann allo IAS di Princeton. Il progetto EDVAC perse così i suoi padri prima ancora di essere realizzato. Tuttavia, le caratteristiche peculiari dell'EDVAC contenute nel *First Draft* - che furono oggetto anche di controversie giudiziarie - influenzarono e finirono per trasformare profondamente la macchina che esso avrebbe dovuto superare, il suo antecedente diretto, l'ENIAC. Nell'aprile del 1948 questo calcolatore fu riconfigurato secondo il nuovo paradigma e, per la maggior parte della sua vita operativa, fino alla disattivazione nella notte del 2 ottobre 1955, funzionò come una macchina a programma memorizzato [Cfr. Haigh et al. 2016, p. 153].

La Moore School era stato il fulcro dello sviluppo delle nuove grandi macchine computazionali. "Durante questo periodo, moltissimi visitatori arrivarono alla Moore School e ripartirono per diffondere il verbo del nuovo mondo del calcolatore." [Goldstine, 1980, p.211 trad. mia]⁴⁹⁷. Venuti meno i protagonisti dei progetti PX e PY (come erano chiamati ENIAC ed EDVAC) questa perse il suo ruolo centrale. Tuttavia il centro dell'Università della Pennsylvania ebbe "ebbe una rinascita nell'estate del 1946" [Goldstine, 1972, p.241 trad. mia]⁴⁹⁸.

La Moore School, con un finanziamento di 7000 dollari da parte del Dipartimento dell'Ordinance dell'esercito degli Stati Uniti, organizzò, dall' 8 giugno fino al 30 agosto 1946, una summer school intitolata *Theory and Techniques for Design of Electronic Digital Computers*, nota poi come *Moore School Lectures* a cui parteciparono, in qualità di relatori, tra gli altri, moltissimi dei protagonisti dello sviluppo dei calcolatori incontrati sinora: J. Presper Eckert, John Mauchly, Herman Goldstine, John von Neumann, Arthur Burks, Howard Aiken, George Stibitz, Jan Rajchman. Un vero "*Who's Who* del computing" [Campbell-Kelly, Williams, 1985, p.xv]⁴⁹⁹.

⁴⁹⁷ "During this period very many visitors came to the Moore School and left to spread the word about the new world of the computer."

⁴⁹⁸ "The Moore School had a resurgence in the summer of 1946".

⁴⁹⁹ Le trascrizioni delle lezioni tenute durante le *Moore School Lectures* si trovano in ed. Campbell-Kelly, Williams, 1985.

Le Moore School Lectures furono “un vettore cruciale per la diffusione del *concetto di programma memorizzato*” [Haigh et al., 2016, p. 154, trad. mia]⁵⁰⁰. Con l’ENIAC ancora in fase di completamento e l’EDVAC soltanto su carta, i principi architetturali di quest’ultimo costituirono il centro degli interventi dei relatori e catalizzarono l’attenzione dell’uditorio.

A testimonianza di ciò vi è, ad esempio, l’intervento di uno degli ideatori della prima macchina della Moore School, J. Presper Eckert. In contrasto con lo sconcertante intreccio di cavi, fili e prese da configurare nei pannelli dell’ENIAC, egli affermava: “nell’EDVAC non ci saranno cavi, né spinotti, e pochi interruttori. Useremo semplicemente la memoria per conservare elettronicamente le informazioni e per inviare, dalla memoria ai circuiti di controllo, quei pezzi di informazione relativi alla programmazione, al fine di sequenziare la macchina in modo che esegua le sue varie operazioni” [Eckert, 1985, p. 114, trad. mia]⁵⁰¹.

La memoria, in termini di prestazioni e costi, è più volte indicata come l’architrave del design del nuovo paradigma di calcolatore (Eckert stava perfezionando il sistema a linee di ritardo); infatti, rispetto alla frammentazione in diversi tipi e configurazioni di memoria - con conseguente perdita di capacità di immagazzinamento ed efficienza - Eckert affermava:

Nell’EDVAC, intendiamo utilizzare un’unica forma di memoria interna per tutte le operazioni interne. In altre parole, se riusciremo a produrre un tipo di memoria sufficientemente economico, [...], non vi è alcuna ragione per cui la stessa forma di memoria non possa essere usata per tutti e tre i tipi di memoria. Oltre a semplificare la costruzione della macchina, questa scelta elimina per il progettista il problema di dover trovare il giusto equilibrio tra i vari tipi di memoria, demandando invece il compito all’utente, che è molto più in grado di effettuare una suddivisione appropriata, poiché si trova ogni volta a trattare con un problema specifico e non con una classe di problemi [...] In effetti, nell’uso dell’ENIAC, ci si trova nella difficile condizione di dover adattare il problema alla macchina, anziché avere una macchina che si adatti al problema [Eckert, 1985, p.112 trad. mia]⁵⁰².

⁵⁰⁰ “Historians have credited the summer school, often called simply the “Moore School Lectures,” as a crucial vector for the spread of the “stored program concept.””.

⁵⁰¹ “However, in the EDVAC there will be no cords, no plugs, and few switches. We are simply going to use the memory to hold the information electronically and to feed those pieces of information which relate to programming from the memory, into the control circuits in order to sequence the machine so that it will perform its various operations”.

⁵⁰² “The purpose of making this estimate of the internal memory capacity of the ENIAC is that, in the EDVAC, it is our intention to use only one form of internal memory to provide for all internal operations. In other words, if we can produce a form of memory cheap enough [...] there is no reason why the same form of memory cannot be utilized for all three types of memory. Aside from simplifying the construction of the machine, this move eliminates for the designer the problem of attempting to find the proper balance between the various types of the memory, and gives the problem to the user who is much more capable to make a proper subdivision since he is, at any time,

A sostenere queste scelte di progettazione, oltre allo stato dell'arte tecnologico disponibile (che si trattasse di valvole termoioniche, tubi catodici o delay-line al mercurio), non vi era soltanto la considerazione costi/benefici in termini di performance. A favore dell'architettura di von Neumann giocava anche una precisa presa di posizione sulla natura del cervello come elaboratore sequenziale di informazioni, in perfetta coerenza con la metafora cibernetica adottata nel *First Draft of a Report on the EDVAC*: "Il cervello umano non pensa in diversi canali paralleli allo stesso tempo; di solito elabora le cose passo dopo passo. Pertanto, sotto ogni aspetto, si è ritenuto estremamente auspicabile costruire la macchina in modo che compia un solo passaggio per volta" [Eckert, 1946, p. 114, trad. mia]⁵⁰³

Programma memorizzato; miglioramento dell'hardware; architettura seriale; analogia funzionale tra gli organi della macchina e il cervello umano: furono questi i principi che la momentanea sospensione della diaspora dalla Moore School contribuì a diffondere e che gli uditori del ciclo di lezioni dell'estate del 1946 - tra cui Maurice Wilkes - fecero propri, disseminandoli nei centri in cui si progettavano e costruivano calcolatori.

"Per il minuscolo gruppo di ingegneri (me compreso) destinati a costruire i primi calcolatori a programma memorizzato," ricorda Wilkes, "i dettagli del design dell'ENIAC [...] suscitavano poco interesse" [Wilkes, 2006, trad. mia]⁵⁰⁴.

Tuttavia, per quanto potesse apparire obsoleto agli apostoli del programma memorizzato rispetto all'*hype* suscitato dai principi dell'EDVAC, l'ENIAC restava "la macchina più veloce esistente, e quindi ancora di grande interesse per i gruppi che dovevano affrontare calcoli complessi". Tra questi vi era quello impegnato ad affrontare la più ardua sfida computazionale di tutte: lo studio delle reazioni nucleari a catena. "Los Alamos aveva accumulato un arretrato di bisogni computazionali su larga scala, ai quali si sarebbe potuto far fronte solo quando il computer in costruzione presso l'Institute for Advanced Study fosse stato

dealing with a specific problem and not a class of problems [...] Actually, in using the ENIAC, one is placed in the difficult position of attempting to fit the problem to the machine rather than having the machine fit the problem".

⁵⁰³ "The human brain does not think in several parallel channels at the same time; it usually thinks these things out step by step. Therefore, in all ways, it is found exceedingly desirable to build the machine so that only single steps are performed at any time".

⁵⁰⁴ "However, for the very reasons outlined, the details of the ENIAC design and of its "second life" at Aberdeen were of little interest to the small group of people who were informed about the EDVAC developments. This was especially true of the minute group of engineers (including myself) who were destined to build the first stored program computers-that is, Eckert-von Neumann computers".

operativo - cosa che, nei primi mesi del 1947, anche nelle stime più ottimistiche, non si prevedeva accadesse prima di diversi anni” [Haigh et al., 2016, p. 156, trad. mia]⁵⁰⁵

L'unica opzione a breve termine per far fronte alla mole di calcoli relativi agli ordigni nucleari rimaneva l'ENIAC, le cui potenzialità erano ben note a von Neumann. Il matematico ungherese intendeva utilizzarlo - dopo i primi contatti di Los Alamos con la macchina attraverso Frankel e Metropolis - per sperimentare un nuovo metodo di analisi statistica “elaborato al fine di modellizzare stocasticamente il problema della diffusione dei neutroni” [Israel, Millán Gasca, 2008, p. 211], il nascente metodo Monte Carlo. “Entro il marzo del 1947, von Neumann aveva chiesto ai suoi stretti collaboratori Herman e Adele Goldstine, entrambi veterani dei tempi della Moore School dell'ENIAC, di studiare la ‘configurazione’ dell'ENIAC per il Monte Carlo” [Haigh et al., 2016, p. 156, trad. mia]⁵⁰⁶.

Dei due progetti interconnessi - la conversione dell'ENIAC secondo i principi del calcolatore a programma memorizzato e l'implementazione del metodo Monte Carlo sulla macchina originale della Moore School - ci soffermeremo anche su quest'ultimo, seguendo l'analisi di P. Galison contenuta nel saggio *Computer Simulations and the Trading Zone* [Galison, 1996].

L'applicazione del metodo Monte Carlo sull'ENIAC è stata resa possibile dalla nuova configurazione architeturale e dalla trasformazione dei paradigmi di programmazione⁵⁰⁷. Alcune conseguenze, sottolineate da Galison, tuttavia risultano particolarmente rilevanti per il discorso qui sviluppato: l'emergere di una *trading zone* tra diversi saperi (matematica, fisica, scienze della computazione, ingegneria) e tra differenti ambiti (accademico, industriale, militare); la formazione di una *lingua franca* come espressione di tale scambio; e l'apertura di un nuovo dominio, quello della *simulazione*, con le sue ricadute epistemologiche e ontologiche. Su questi cardini si innestarono gli sviluppi successivi della computazione e della sua interazione con l'essere umano, con calcolatori ormai concepiti come strumenti scientifici, commerciali e mediatori con quell'universo dell'informazione che abbiamo quantificato nelle pagine precedenti.

⁵⁰⁵ “Los Alamos had a backlog of large-scale computing needs to which the computer being built at the Institute of Advanced Study would be applied at the earliest possible date. However, in early 1947 even the most optimistic estimates assumed that that machine would not be operational for several years”.

⁵⁰⁶ “By March of 1947, von Neumann had asked his close collaborators Herman and Adele Goldstine, both veterans of ENIAC's Moore School days, to investigate the “set-up” of ENIAC for Monte Carlo”.

⁵⁰⁷ Il processo di conversione di ENIAC al nuovo paradigma di memoria e programmazione è analizzato in Haigh et al., 2016 pp.156-171.

“L’idea di fondo del metodo Monte Carlo è quella di modellare un problema complesso in modo stocastico, così da renderne possibile il calcolo della soluzione” [Aspray, 1990, pp. 110–111, trad. mia]⁵⁰⁸. L’ideatore di questa metodologia statistica è considerato, insieme a John von Neumann, Stanislaw Ulam. Fu lui il primo a renderla pubblica, insieme a Nicholas Metropolis, con l’articolo del 1949 *The Monte Carlo Method* [Metropolis, Ulam 1949].

Matematico di origine polacca, Stanislaw Ulam aveva conosciuto von Neumann a Varsavia, e fu proprio quest’ultimo a invitarlo all’Institute for Advanced Study (IAS) di Princeton. Alle soglie della Seconda guerra mondiale lasciò l’Europa ed emigrò definitivamente negli Stati Uniti nel 1939. Nel 1943 raggiunse Los Alamos, proveniente dall’Università del Wisconsin. Qui lavorò inizialmente nella Divisione Teorica (*T-Division*), diretta da Hans Bethe (1906–2005), e in seguito nella sezione guidata da Enrico Fermi (1901–1954), la *F-Division*.⁵⁰⁹

A guerra conclusa, Ulam accettò una cattedra all’Università della California, ma poco dopo si ammalò gravemente di encefalite virale, una malattia che mise in pericolo la sua vita. Persa l’assicurazione sanitaria governativa e non avendo ancora assunto il nuovo incarico, la sua situazione appariva critica. Fu grazie all’interessamento di von Neumann che ottenne un nuovo invito a rientrare a Los Alamos [cfr. Ulam, 1976, pp. 174-185; Dyson, 2012, p. 219, trad. it.].

L’ideazione e lo spirito del metodo Monte Carlo vengono da lui ricondotti al periodo della convalescenza, quando gli era stato proibito qualsiasi sforzo mentale e si distraeva giocando solitari con le carte. Ricorda Ulam:

L’idea di quello che fu in seguito chiamato metodo Monte Carlo mi venne mentre stavo giocando a solitario durante una mia malattia. Notai che poteva essere molto più pratico farsi un’idea della probabilità di successo di una partita (come quella al Canfield, o altri giochi in cui l’abilità del giocatore non è rilevante) semplicemente disponendo le carte, o sperimentando il processo e osservando quale proporzione si conclude con successo, anziché tentare di calcolare tutte le possibilità combinatorie, il cui numero cresce esponenzialmente al punto che, salvo casi molto elementari, non esiste un modo realistico per stimarlo [Ulam, 1976, pp.196-197 trad.mia]⁵¹⁰.

⁵⁰⁸ “The underlying idea of the Monte Carlo method is to model a complex problem stochastically in a way that admits of a solution's being calculated”.

⁵⁰⁹ La vicenda biografica di Ulam è narrata dallo stesso matematico in Ulam, 1976.

⁵¹⁰ “The idea for what was later called the Monte Carlo method occurred to me when I was playing solitaire during my illness. I noticed that it may be much more practical to get an idea of the probability of the successful outcome of a solitaire game (like Canfield or some other where the skill of the player is not important) by laying down the cards, or experimenting with the process and merely noticing what proportion comes out successfully, rather than

In un misto di sorpresa e sgomento di fronte al “limiti del pensiero razionale tradizionale”, il matematico polacco si rese conto che “in un problema sufficientemente complicato, il campionamento reale è meglio dell’esame di tutte le catene di possibilità” [Ulam, 1976, p.197 trad. mia]⁵¹¹.

“Calcolare la probabilità di esito favorevole in un gioco di solitario (intendiamo qui solo giochi in cui l’abilità non gioca alcun ruolo),” scriverà Ulam nella prima pubblicazione del metodo - che deve il suo nome alla capitale del gioco d’azzardo, in chiara allusione alla dimensione probabilistica coinvolta - “è un compito del tutto intrattabile”. In questo tipo di “problemi combinatori che si presentano sia nella matematica pura sia nelle scienze applicate”, di cui il gioco di carte è un esempio illustrativo, “la procedura pratica è quella di produrre un gran numero di esempi di uno specifico gioco e poi esaminare la proporzione relativa di successi” [Metropolis, Ulam, 1946, trad. mia]⁵¹².

Ulam si rese conto che questo modo di procedere statistico poteva essere applicato anche ai problemi legati alle reazioni delle particelle nei processi termonucleari. “Mi venne quindi in mente che questo poteva valere anche per tutti i processi che coinvolgono ramificazioni di eventi, come nella produzione e moltiplicazione successiva di neutroni in un materiale contenente uranio o altri elementi fissili” [Ulam, 1976, p. 197, trad. mia]⁵¹³. Partendo da alcune costanti fisiche e dalle caratteristiche elementari di un neutrone - come posizione, velocità o direzione di moto - era possibile calcolarne il destino: se urtando un altro avrebbe cambiato direzione o velocità, se sarebbe uscito dall’arma oppure se avrebbe causato una fissione liberando a sua volta altri neutroni⁵¹⁴.

to try to compute all the combinatorial possibilities which are an exponentially increasing number so great that, except in very elementary cases, there is no way to estimate it”.

⁵¹¹ “This is intellectually surprising, and if not exactly humiliating, it gives one a feeling of modesty about the limits of rational or traditional thinking. In a sufficiently complicated problem, actual sampling is better than an examination of all the chains of possibilities”.

⁵¹² “An analogous situation exists in problems of combinatorial analysis and of the theory of probabilities. To calculate the probability of a successful outcome of a game of solitaire (we understand here only such games where skill plays no role) is a completely intractable task. On the other hand, the laws of large numbers and the asymptotic theorems of the theory of probabilities will not throw much light even on qualitative questions concerning such probabilities. Obviously the practical procedure is to produce a large number of examples of any given game and then to examine the relative proportion of successes. The “solitaire” is meant here merely as an illustration for the whole class of combinatorial problems occurring in both pure mathematics and the applied sciences”.

⁵¹³ “It occurred to me then that this could be equally true of all processes involving branching of events, as in the production and further multiplication of neutrons in some kind of material containing uranium or other fissile elements”.

⁵¹⁴ Metropolis fa notare - citando la testimonianza del fisico Emilio Segrè (1905–1989), Premio Nobel per la Fisica nel 1959 - come Enrico Fermi (1901–1954) avesse indipendentemente sviluppato il metodo Monte Carlo già nei

Era tuttavia la scala stessa del problema a renderlo intrattabile con i metodi probabilistici ordinari. “Il problema è sapere che cosa farà una successione e ramificazione di forse centinaia di migliaia o milioni di eventi. Si possono scrivere equazioni differenziali o equazioni integrali-differenziali per i ‘valori attesi’, ma risolverle - o anche solo farsi un’idea approssimativa delle proprietà della soluzione - è tutta un’altra faccenda” [Ulam, 1976, p. 197, trad. it.]⁵¹⁵.

Ulam discusse con von Neumann di questi schemi probabilistici nelle loro conversazioni a Los Alamos nel 1946. Dopo una discussione che il matematico polacco ricordò come “eccezionalmente lunga”, avvenuta durante un viaggio in macchina da Los Alamos a Lamy - snodo ferroviario vicino al centro di ricerca nucleare, dove lo attendeva il treno per Princeton - von Neumann si convinse della validità del metodo [cfr. Ulam, 1976, p. 199].

L’11 marzo 1947 il matematico ungherese scrisse quindi a Robert D. Richtmyer (1910–2003), allora capo della Divisione Teoretica di Los Alamos, illustrandogli le sue idee sull’impiego del metodo Monte Carlo. In questa lettera von Neumann prese in considerazione una “geometria sfericamente simmetrica” [Richtmyer, von Neumann, 1947, p. 751 trad. mia]⁵¹⁶. A partire da questo nucleo sferico di materiale fissile, circondato da un guscio assorbente [*tamper*], von Neumann propose di considerare la distribuzione iniziale di neutroni in base alla posizione e alla velocità, e seguirne l’evoluzione ignorandone gli sviluppi del comportamento

primi anni ’30, durante i suoi studi sulla moderazione dei neutroni a Roma. Sebbene non ci siano pubblicazioni a riguardo, Fermi usava tecniche di campionamento statistico per risolvere problemi fisici con i mezzi allora disponibili, come una semplice addizionale meccanica. Secondo Segrè, Fermi amava stupire i colleghi romani con previsioni sperimentali “troppo esatte per essere vere”, derivate dai calcoli che eseguiva nottetempo. In suo onore, nel 1947 a Los Alamos fu progettato un dispositivo meccanico su rotaie per simulazioni di diffusione statistica dei neutroni, che fu soprannominato affettuosamente FERMIAC [Metropolis, 1987, p.128]. “Il carrello Monte Carlo, o FERMIAC, fu inventato da Enrico Fermi e costruito da Percy King. I tamburi del carrello venivano impostati in base al materiale attraversato e a una scelta casuale tra neutroni veloci e lenti. Un altro numero casuale veniva usato per determinare la direzione del moto, e un terzo veniva selezionato per stabilire la distanza fino alla successiva collisione. Il carrello veniva poi mosso manualmente su un disegno in scala bidimensionale del dispositivo nucleare o dell’assemblaggio del reattore in esame. Durante il movimento, il carrello tracciava un percorso, fermandosi ogni volta che si attraversava un confine tra materiali per aggiornare le impostazioni dei tamburi. Questo rudimentale computer fu utilizzato per circa due anni per determinare, tra le altre cose, la variazione nel tempo della popolazione di neutroni in numerosi tipi di sistemi nucleari” [“The Monte Carlo trolley, or FERMIAC, was invented by Enrico Fermi and constructed by Percy King. The drums on the trolley were set according to the material being traversed and a random choice between fast and slow neutrons. Another random digit was used to determine the direction of motion, and a third was selected to give the distance to the next collision. The trolley was then operated by moving it across a two-dimensional scale drawing of the nuclear device or reactor assembly being studied. The trolley drew a path as it rolled, stopping for changes in drum settings whenever a material boundary was crossed. This infant computer was used for about two years to determine, among other things, the change in neutron population with time in numerous types of nuclear systems”, Metropolis, 1987, p.129 trad. mia]

⁵¹⁵ “The elementary probabilities for each of these possibilities are individually known, to some extent, from the knowledge of the cross sections. But the problem is to know what a succession and branching of perhaps hundreds of thousands or millions will do. One can write differential equations or integral differential”.

⁵¹⁶ “Spherically symmetric geometry”.

in relazione a variabili radioattive o idrodinamiche [Cfr. Eckhardt, 1987, pp.133-134; Metropolis, 1987, p.127].

L'idea era "quella di ricostruire la storia di un dato neutrone", e questa storia "è analoga a una singola partita di solitario" [Eckhardt, 1987, p.133 trad. mia]⁵¹⁷. Il metodo Monte Carlo sembrava incarnare quella stessa apparente impossibilità che, nella storia della scienza, era stata prerogativa esclusiva del demone di Maxwell - non a caso evocato in apertura dell'articolo con cui Ulam e Metropolis conferirono statuto pubblico al metodo [Metropolis, Ulam, 1949] - ossia "un modo per seguire effettivamente un sistema fisico ai livelli più elementari" [Dyson, 2012, p.253, trad. it.]

Per tracciare la storia evolutiva di un singolo neutrone e ricavarne un campione significativo di traiettorie, occorre considerare migliaia - ma non la totalità - dei possibili casi, ciascuno sviluppato lungo una singola linea di eventi.

A ogni stadio viene presa una decisione, o meglio una sequenza di decisioni, fondate su probabilità statistiche determinate dai fattori fisici e geometrici in gioco. Le prime due scelte si compiono già al tempo $t = 0$, quando si assegna al neutrone una certa velocità e una posizione iniziale. Nel momento in cui il neutrone attraversa un confine materiale, entrano in gioco i parametri e le caratteristiche del nuovo mezzo. In questo modo si costruisce la genealogia di un singolo neutrone. L'intero procedimento viene poi ripetuto per molti altri neutroni, fino a ottenere un quadro complessivo statisticamente significativo [Metropolis, 1987, p. 127 trad. mia]⁵¹⁸.

Per prendere queste decisioni e generare un campione statistico, era necessario partire da dei *numeri casuali*. Ulam ricorda come la sua proposta iniziale fu quella di "assumere diverse centinaia di cinesi da Taiwan, radunarli su una nave, farli sedere ciascuno con un abaco, o anche solo con carta e matita, e fargli generare i numeri casuali tramite qualche processo fisico reale,

⁵¹⁷ "The idea then was to trace out the history of a given neutron; In von Neumann's formulation of the neutron diffusion problem, each neutron history is analogous to a single game of solitaire".

⁵¹⁸ "At each stage a sequence of decisions has to be made based on statistical probabilities appropriate to the physical and geometric factors. The first two decisions occur at time $t = 0$, when a neutron is selected to have a certain velocity and a certain spatial position. The next decisions are the position of the first collision and the nature of that collision. If it is determined that a fission occurs, the number of emerging neutrons must be decided upon, and each of these neutrons is eventually followed in the same fashion as the first. If the collision is decreed to be a scattering, appropriate statistics are invoked to determine the new momentum of the neutron. When the neutron crosses a material boundary, the parameters and characteristics of the new medium are taken into account. Thus, a genealogical history of an individual neutron is developed. The process is repeated for other neutrons until a statistically valid picture is generated".

come il lancio di dadi. Qualcuno poi avrebbe raccolto i risultati e sommato le statistiche per ottenere le risposte finali” [Ulam, 1976, trad. mia⁵¹⁹].

Rispetto a questa boutade scherzosa - che rende in modo ironico ed efficace il passaggio dai compiti computazionali svolti con carta e matita alla loro elaborazione tramite calcolatore - e muovendo dalla ferma convinzione che “chiunque prenda in considerazione metodi aritmetici per generare cifre casuali si trova, naturalmente, in uno stato di peccato” [von Neumann, 1951, p. 758, trad. mia]⁵²⁰, von Neumann propose dei metodi algoritmici per la generazione di cifre casuali come quello delle “cifre centrali” [*middle-square digits*]. “In esso, un numero intero arbitrario di n cifre viene elevato al quadrato, producendo un numero di $2n$ cifre. Da questo prodotto si ricava un nuovo intero estraendo le n cifre centrali. Il processo viene ripetuto più e più volte, formando una catena di interi le cui proprietà sono state studiate a fondo” [Metropolis, 1987, p.127 trad. mia]⁵²¹.

È evidente la complessità - e la mole imponente - di calcoli richiesta per generare numeri casuali e, a partire da essi, le storie campione necessarie all'applicazione del metodo Monte Carlo. Ulam osservava tuttavia, con sottile ironia, che “per caso, proprio allora stavano nascendo le macchine calcolatrici, e questo era un ambito perfetto per il calcolo automatico”. [Ulam, 1976, p.197 trad. mia]⁵²².

⁵¹⁹ “As a joke I proposed to make Monte Carlo calculations by hiring several hundred Chinese from Taiwan, gather them on a boat, have each one sit with an abacus, or even just pencil and paper, and make them produce the random numbers by some actual physical process like throwing dice. Then someone would collect the results, and total the statistics into single answers”.

⁵²⁰ “Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin”. J. von Neumann pubblicò le sue idee sulla generazione randomizzata di cifre nel 1951. Secondo il matematico ungherese non esistono cifre veramente casuali: esistono solo “ricette” (*cooking recipes*), come il suo metodo delle cifre centrali, per generarle. George Dyson osserva come “il successo del metodo Monte Carlo fu accompagnato da un'improvvisa domanda di numeri casuali affidabili, che scarseggiavano. [...] Nell'aprile 1947 il progetto RAND dell'aeronautica americana (progenitore della RAND Corporation), per cui von Neumann lavorava come consulente a Santa Monica, decise di costruire una roulette elettronica e di compilare un elenco di un milione di numeri casuali, dapprima disponibile sotto forma di schede perforate e in seguito ampliato e pubblicato in un libro” [Dyson, 2012, p. 228, trad. it.] John von Neumann scrisse, insieme a Nicholas Metropolis e al matematico del *National Bureau of Standards* George Reitwiesner (1914–1969), uno studio sulle prime 2000 cifre randomizzate calcolate con l'ENIAC [Metropolis et al., 1950]. Un'eco letteraria della “fame” di numeri casuali si trova nel romanzo di fantascienza dello scrittore polacco Stanisław Lem (1921–2006), che affrontò spesso nella sua narrativa e nei suoi saggi i temi della cibernetica, della robotica e del rapporto tra essere umano e computazione. In *La voce del Padrone* (1968), gli eventi, inseriti in una cornice verosimile di progetto scientifico-militare segreto, sono innescati da un segnale stellare che inizialmente uno dei personaggi interpreta come una possibile fonte di numeri casuali da rivendere con buon profitto [Lem, 1968, pp. 50–51, trad. it.].

⁵²¹ “Here, an arbitrary n -digit integer is squared, creating a $2n$ -digit product. A new integer is formed by extracting the middle n -digits from the product. This process is iterated over and over, forming a chain of integers whose properties have been extensively studied”.

⁵²² “It so happened that computing machines were coming into existence, and here was something suitable for machine calculation”. A proposito del rapporto tra numeri casuali e calcolatori, secondo von Neumann il migliore tra i metodi disponibili - esclusi quelli puramente aritmetici e ritenute inaccettabili le fonti fisiche dirette, come le sequenze ottenute da fenomeni casuali naturali (es. decadimenti radioattivi, il lancio di un dado o rumore

Nella sopra menzionata lettera a Richtmyer, riferendosi ai “metodi statistici per risolvere problemi di diffusione e moltiplicazione neutronica, secondo il principio suggerito da Stan Ulam”, von Neumann mette in evidenza come “l’approccio statistico è molto adatto a un trattamento digitale” [Richtmyer, von Neumann, 1947, p. 751 trad. mia]⁵²³

Non si può non osservare come anche in questo caso la macchina digitale - analogamente al calcolo delle serie statistiche alla base della cibernetica o ai calcolatori crittoanalitici di Bletchley Park - diventi lo strumento che “solidifica” la complessità probabilistica e rende possibile il suo governo. La macchina designata per il trattamento digitale dei calcoli probabilistici delle reazioni neutroniche fu l’ENIAC: “Sono abbastanza certo che il problema [...] nella sua forma digitale, sia ben adatto all’ENIAC” [Richtmyer, von Neumann, 1947, p. 752 trad. mia]⁵²⁴.

Nella stessa missiva von Neumann elabora una stima dei tempi di calcolo:

La mia stima attuale (preliminare) è la seguente: supponiamo che un problema di criticità richieda di seguire 100 neutroni primari attraverso 100 collisioni (del neutrone primario o dei suoi discendenti) per ciascun neutrone primario. Risolvere un singolo problema di criticità dovrebbe quindi richiedere circa 5 ore [...] Una configurazione comune dell’ENIAC andrà bene per tutti i problemi di criticità. Per passare da un problema di questa categoria a un altro, sarà necessario impostare solo nuovi valori numerici in uno degli organi di “tabelle di funzioni” dell’ENIAC [Richtmyer, von Neumann, 1947, p. 752 trad. mia]⁵²⁵.

Il matematico ungherese fornisce, inoltre, uno schema di schede di programmazione [*computer sheet*] che dovevano costituire la base della configurazione di ENIAC ai calcoli del Monte Carlo. Prosegue von Neumann: “Non posso ancora affermarlo con certezza, ma mi

elettronico), per l’impossibilità di ripetere i controlli durante i calcoli - “la soluzione meno peggiore sarebbe produrre cifre casuali tramite un meccanismo fisico e registrarle, lasciando che la macchina le legga quando necessario. A questo punto, però, ci siamo messi nella condizione di dover utilizzare la parte più debole delle macchine attualmente progettate: l’organo di lettura” [“The next best thing would be to produce random digits by some physical mechanism and record them, letting the machine read them as needed. At this point we have maneuvered ourselves into using the weakest portion of presently designed machines-the reading organ”, von Neumann 1951 trad. mia].

⁵²³ “I have been thinking a good deal about the possibility of using statistical methods to solve neutron diffusion and multiplication problems, in accordance with the principle suggested by Stan Ulam [...] The statistical approach is very well suited to a digital treatment”.

⁵²⁴ “I am fairly certain that the problem of (1), in its digital form, is well suited for the ENIAC”.

⁵²⁵ “My present (preliminary) estimate is this: Assume that one criticality problem requires following 100 primary neutrons through 100 collisions (of the primary neutron or its descendants) per primary neutron. Then solving one criticality problem should take about 5 hours. [...] A common set-up of the ENIAC will do for all criticality problems. In changing over from one problem of this category to another one, only a new numerical constants will have to be set anew on one of the “function table” organs of the ENIAC”.

sembra molto probabile che le istruzioni fornite in questo "computer sheet" non superino la capacità "logica" dell'ENIAC." [Richtmyer, von Neumann, 1947, p. 758 trad. mia]⁵²⁶

L'idea di von Neumann era di impiegare le schede perforate per rappresentare il destino dei neutroni. "Non vedo alcun motivo per assegnare più di, diciamo, 7 posizioni a ciascuna delle 5 caratteristiche, né più di, diciamo, 5 posizioni a ciascuna delle 7 variabili casuali" [Richtmyer, von Neumann, 1947, p. 757 trad. mia]⁵²⁷

Il comportamento futuro di ciascun neutrone poteva dipendere da fenomeni quali dispersione, assorbimento, fuoriuscita o fissione. I dati registrati sulla scheda, indispensabili per determinarne statisticamente il destino, includevano la zona dell'agglomerato sferico in cui si verificava l'interazione, l'istante dell'evento, la velocità della particella in quel momento, l'angolo polare della traiettoria, la distanza dal centro della sfera e il numero di neutroni rappresentati dalla scheda stessa [Cfr. Dyson, 2012, pp. 225–226, trad. it.].

In questo modo, secondo von Neumann, verranno utilizzate soltanto 70 posizioni sulla *punched card*, per cui "la comune scheda perforata da 80 colonne avrà 10 posizioni libere da destinare a eventuali indicizzazioni aggiuntive, eccetera, se lo si desidera. Il processo computazionale dovrebbe quindi essere organizzato in modo da produrre la scheda C' del "nuovo" neutrone, o meglio da 1 a 4 schede - C', C", C"', C''' - a seconda della storia effettiva del neutrone" [Richtmyer, von Neumann, 1947, p. 756 trad. mia]⁵²⁸

Nonostante la fiducia di von Neumann nelle capacità logico-computazionali dell'ENIAC "classico" e delle sue tavole di funzioni - insiemi di interruttori che ne comandavano direttamente il funzionamento - procedeva parallelamente il lavoro di adattamento del calcolatore al paradigma dell'EDVAC. Lo stesso von Neumann si dichiarava convinto che le macchine "post-ENIAC" sarebbero state in grado di affrontare con maggiore efficacia i problemi in questione [Cfr. Richtmyer, von Neumann, 1947, p.753].

Al "barattolo di vermi" di cavi, pannelli e spinotti della configurazione originale si andava sostituendo una struttura fissa, a programma memorizzato, con una capacità di 1800 istruzioni

⁵²⁶ "I cannot assert this with certainty yet, but it seems to me very likely that the instructions given on this "computing sheet" do not exceed the "logical" capacity of the ENIAC".

⁵²⁷ "Each neutron is represented by a card *C* which carries its characteristics *i*, *r*, *s*, *v*, *t*, and also the necessary random values λ , μ , ν , ρ' , ρ'' , ρ''' . I can see no point in giving more than, say, 7 places for each one of the 5 characteristics, or more than, say, 5 places for each of the 7 random variables".

⁵²⁸ "At any rate, even in this way only 70 entries are consumed, and so the ordinary 80-entry punchcard will have 10 entries left over for any additional indexings, etc., that one may desire. The computational process should then be so arranged as to produce the card *C'* of the "new" neutron, or rather 1 to 4 such cards *C'*, *C''*, *C'''*, *C''''* (depending on the neutron's actual history, cf. above). Each card, however, need only be provided with the 5 characteristics of its neutron".

tratte da un vocabolario di circa 60 operazioni aritmetiche e logiche [Metropolis, 1987, p.128]⁵²⁹.

Per quanto riguarda la conversione dell'ENIAC, il merito spetta ad Adele Goldstine (1920–1964) - all'epoca poco meno che trentenne e moglie di Herman Goldstine - la quale propose di sovrapporre all'hardware della macchina, ritenuto da Presper Eckert sufficientemente flessibile [Cfr. Eckert, 1980, pp. 526–530], le strutture delineate nel *First Draft*, ossia gli *organi* del calcolatore concepiti ad un alto livello di astrazione logico-cibernetica secondo il modello dell'EDVAC.

L'approccio prevedeva di ripartire i venti accumulatori dell'ENIAC in gruppi funzionali: un sistema aritmetico, con un accumulatore centrale (il 15), uno ausiliario (il 13) e registri specializzati per operazioni complesse; un sistema di controllo per il prelievo e la decodifica delle istruzioni; e una programmazione distribuita, di tipo *special-purpose*, tramite cablaggi e interruttori, secondo la vecchia struttura decentralizzata dell'ENIAC [Cfr. Haigh et al., 2016, pp. 159–160].

Ad un'altra donna, la seconda moglie di von Neumann, Klara (1911-1963) si deve attribuire un ruolo fondamentale nel lavoro di programmazione per l'implementazione del Monte Carlo su ENIAC svolto a Princeton⁵³⁰.

Intanto, mentre il team di Princeton lavorava per trasformare “il piano di calcolo originario nella progettazione di un programma pienamente articolato per l'ENIAC” [Haigh et al., 2014 trad. mia]⁵³¹, questo era guidato dallo studio svolto dai mariti delle due - fin troppo, e a torto, misconosciute - programmatrici: Herman Goldstine e John von Neumann, che ne avevano espresso i principi nel loro *Planning and Coding Problems for an Electronic Computing Instrument* [Goldstine, von Neumann, 1947].

Al centro di questo lavoro vi era il concetto di diagramma di flusso [*flow diagram*]. “La codifica inizia con il disegno dei diagrammi di flusso” affermano nel lavoro i due matematici: “questa è la fase dinamica o macroscopica della codifica” [Goldstine, von Neumann, 1947, p.100

⁵²⁹ “This mode would have a capacity of 1800 instructions from a vocabulary of about 60 arithmetical and logical operations. The previous method of programming might be likened to a giant plugboard, that is to say, to a can of worms”.

⁵³⁰ Per quanto riguarda la vicenda biografica di Klára (Klari) von Neumann, nata Dán, e il suo impegno nei progetti di Los Alamos e in Pennsylvania sulla programmazione - intrecciato con la relazione sentimentale con von Neumann, al punto che il lavoro con i calcolatori finì per salvare il loro matrimonio - si rimanda a Dyson, 2012, pp. 203–213; 215–218; 223–228; 312–317 trad. it. Il ruolo delle due donne, K. Von Neumann e A. Goldstine, nel “miracolo” della riconversione dell'ENIAC, ricostruita storicamente e nel dettaglio tecnico è in Haigh, et al., 2014a.

⁵³¹ “As the team in Princeton worked to transform the original computing plan into the design of a fully articulated program for the ENIAC”.

trad. mia]⁵³². Il diagramma di flusso rappresentava “il cuore di un approccio piuttosto rigoroso per la traduzione delle espressioni matematiche in programmi in linguaggio macchina” [Haigh et al., 2014b, p.47 trad. mia]⁵³³.

Goldstine ricorda come, nell'estate del 1947, si convinse che “questo tipo di diagramma di flusso, come lo chiamammo, potesse essere utilizzato come una notazione logicamente completa e precisa per esprimere un problema matematico, e che ciò fosse, in effetti, essenziale per il compito della programmazione” [Goldstine, 1972, p.266 trad. mia]⁵³⁴.

I diagrammi di flusso, nati attraverso la considerazione logica e la concreta codifica di problemi computazionali da dare in pasto alla macchina, hanno come loro ragion d'essere uno scopo, che è lo stesso Goldstine - matematico, ufficiale e computer scientist - a illustrare:

Lo scopo del diagramma di flusso è fornire un'immagine del movimento dell'organo di controllo mentre si muove attraverso la memoria raccogliendo ed eseguendo le istruzioni che vi trova. Il diagramma di flusso mostra anche gli stati delle variabili in diversi punti chiave del corso del calcolo. Inoltre, indica le formule che vengono valutate. Come ho sottolineato una volta in una lezione sull'argomento, il diagramma di flusso è in realtà una superficie di Riemann con molti fogli, corrispondenti alle varie induzioni che vi entrano [Goldstine, 1972, p.267 trad. mia]⁵³⁵.

L'implementazione del Monte Carlo fu condotta e testata in tre diversi “run” tra il 1948 e il 1949. “All'interno di ciascun run venne utilizzata una diversa evoluzione del codice del programma per studiare una serie di “problemi”, ognuno dei quali simulava una diversa configurazione fisica” [Haigh et al, 2014b, p.54 trad. mia]⁵³⁶.

Il primo *run*, sull'ENIAC configurato da Metropolis per eseguire i nuovi metodi di programmazione nella primavera del 1948 ad Aberdeen, mostrò diversi bug di hardware e di software. Nessuno aveva ancora effettuato il *debug* di un programma scritto secondo il

⁵³² “Coding begins with the drawing of the flow diagrams. This is the dynamic or macroscopic stage of coding”.

⁵³³ “Their reports on “*Planning and Coding Problems for an Electronic Computing Instrument*,” issued in 1947 and 1948, put flow diagrams as the heart of a fairly rigorous approach for the translation of mathematical expressions into machine language programs.”

⁵³⁴ “Then that summer I became convinced that this type of flow diagram, as we named it, could be used as a logically complete and precise notation for expressing a mathematical problem and that indeed this was essential to the task of programming”.

⁵³⁵ “The purpose of the flow diagram is to give a picture of the motion of the control organ as it moves through the memory picking up and executing the instructions it finds there. The flow diagram also shows the states of the variables at various key points in the course’ of the computation. Further, it indicates the formulas being evaluated. As I pointed out once in a lecture on the subject, the flow diagram is really a Riemann surface with many sheets for the various inductions that enter”.

⁵³⁶ “Within each run a different evolution of the program code was used to investigate a number of “problems,” each simulating a different physical configuration”.

paradigma moderno! Furono testate sette situazioni diverse, variando la forma e i materiali delle configurazioni nucleari, per osservare come si comportavano i neutroni in condizioni critiche e subcritiche, tenendo conto delle probabilità di assorbimento, diffusione e fissione in diverse situazioni [Cfr. Haigh et al, 2014b, pp.54-55].

Nel secondo *run*, condotto nell'autunno del 1948, Klara von Neumann ebbe nuovamente un ruolo centrale nella riorganizzazione del codice e nella gestione operazioni di calcolo. A differenza del primo, questo ciclo di simulazioni fu orientato esplicitamente alla progettazione di armi nucleari, incluso lo studio di configurazioni supercritiche con nuclei a idruro di uranio. Uno degli obiettivi raggiunti fu quello di aumentare l'efficienza complessiva del processo. Alcuni passaggi manuali tra un ciclo temporale e il successivo vennero eliminati ed aumentò l'automatizzazione dell'impiego delle schede perforate che rappresentavano l'evoluzione delle particelle. Il terzo *run* ebbe luogo nell'inverno tra il '48 e il 1949. "Klara von Neumann partì per Los Alamos intorno al 1° dicembre, presumibilmente per contribuire all'interpretazione dei risultati del secondo run e per gettare le basi per calcoli futuri" [Haigh et al., 2014b, p.57 trad.mia]⁵³⁷. In questa esecuzione/test del programma vennero scoperti degli errori di calcolo degli idruri. [Cfr. Haigh et al, 2014b, p.57].

Tuttavia, malgrado questi errori, si ebbe una nuova conferma del potenziale della computazione elettronica, nel 1945-46 ENIAC aveva già mostrato l'infattibilità iniziale, confermando i dubbi di Ulam, dei calcoli di Teller sulla *Super*, la bomba all'idrogeno [Cfr. McCartney, 1999, p.103], nella ricerca nucleare.

Monte Carlo fu, con una facile battuta, una scommessa vincente. Il metodo venne implementato anche su un successore dell'ENIAC, il MANIAC, costruito a Los Alamos sotto la guida di Nicholas Metropolis. Progettato secondo i principi dell'architettura di von Neumann, il MANIAC adottava però un proprio sistema di memoria, sviluppato indipendentemente rispetto a quello dell'IAS. In particolare, utilizzava tubi Williams da 2 pollici (5,08 cm) anziché i più grandi tubi da 5 pollici (12,70 cm) adottati a Princeton. Anche il sistema di controllo, il vocabolario delle istruzioni e le periferiche di input/output (gli organi di interfaccia) furono progettati in modo autonomo: il MANIAC non impiegava schede perforate, ma disponeva di un lettore e perforatore fotoelettrico per nastro cartaceo, di un nastro magnetico da 3 pollici (7,62 cm), di una macchina da scrivere, di un tamburo magnetico da 10.000 parole e di una stampante a linea Analex Model 1 [Cfr. Metropolis, 1980, p.461].

⁵³⁷ "Klara von Neumann left for Los Alamos around 1 December, presumably to help interpret the second run results and to lay the groundwork for future calculations".

Anche in questo calcolatore, divenuto operativo il 1952, la tensione era verso l'implementazione della computazione digitale elettronica che rappresenta in un certo senso un *fil rouge* della storia che stiamo sviluppando. Come ricorda il suo artefice principale: "L'inclusione di dispositivi elettromeccanici portò alla sentenza oracolare: "Ogni volta che l'elettromeccanica può essere sostituita dall'elettronica, fallo!" {Metropolis, 1980, p.461 trad. mia]⁵³⁸.

Il MANIAC, battezzato non senza ironia da Nicholas Metropolis come retronimo di *Mathematical Analyzer, Numerical Integrator and Computer*⁵³⁹, fu il calcolatore impiegato per i complessi compiti numerici che condussero, la mattina del 1° novembre 1952 (ora locale del Pacifico), all'esplosione di *Ivy Mike*, la prima bomba all'idrogeno costruita secondo il design Ulam-Teller. Con quell'arma, resa possibile anche dai progressi dei calcolatori digitali e dall'applicazione del metodo Monte Carlo, l'universo digitale cominciò ad espandersi con una portata quasi pari a quella dell'ordigno a cui aveva contribuito a dare forma.

Come fa notare Metropolis nella sua ricostruzione di fine anni 80 del metodo Monte Carlo "È interessante guardare indietro a oltre quarant'anni fa e notare l'emergere, già in una fase piuttosto precoce, della matematica sperimentale, una conseguenza naturale del calcolatore elettronico. Il ruolo del metodo Monte Carlo nel rafforzare tale forma di matematica appare evidente" [Metropolis, 1987, p.129 trad. mia]⁵⁴⁰.

Abbiamo visto come i progetti nucleari di Los Alamos coinvolgessero una serie eterogenea di campi di sapere e pratiche scientifiche, ossia "ciò che all'inizio può apparire come un assemblaggio caotico di discipline e attività: armi termonucleari, bombe A potenziate, gas velenosi, previsione meteorologica, interazioni pione-nucleone, teoria dei numeri, teoria della probabilità, chimica industriale e meccanica quantistica" [Galison, 1996, p. 119, trad. mia]⁵⁴¹. Soltanto pochi - personalità del calibro di Enrico Fermi o John von Neumann - avevano la capacità di padroneggiare tali ambiti e di muoversi in essi con disinvoltura. Ma persino nel caso

⁵³⁸ "The inclusion of electromechanical devices led to the oracular pronouncement, whenever electromechanics can be replaced by electronics, do so!".

⁵³⁹ All'ironia del nome scelto da Metropolis fa il paio quella del fisico, astrofisico e consulente di Los Alamos di origine sovietica George Gamow (1904-1968) - noto per essere stato uno strenuo sostenitore della teoria del Big Bang - secondo cui l'acronimo MANIAC stava per: "*Metropolis And von Neumann Install Awful Computer*" [Metropolis, 1992, p.129]. Un ulteriore approfondimento su MANIAC è in Metropolis, 1992.

⁵⁴⁰ "It is interesting to look back over twoscore years and note the emergence, rather early on, of experimental mathematics, a natural consequence of the electronic computer. The role of the Monte Carlo method in reinforcing such mathematics seems self-evident".

⁵⁴¹ "Instead, I will focus on what appears at first to be a chaotic assemblage of disciplines and activities: thermonuclear weapons, enhanced A-bombs, poison gas, weather prediction, pion-nucleon interactions, number theory, probability theory, industrial chemistry, and quantum mechanics".

di scienziati di questo livello, data la natura dei loro oggetti di studio, “la loro attività comune era centrata attorno al calcolatore” [Galison, 1996, p. 119, trad. mia]⁵⁴². La mole di lavoro computazionale era soverchiante: “se il calcolo fosse stato assegnato a un solo computer (cioè una donna), una simulazione completa di una bomba termonucleare avrebbe richiesto 211,2 anni” [Galison, 1996, p. 133, trad. mia]⁵⁴³.

L’analisi matematica conduceva a equazioni differenziali di estrema complessità, spesso irrisolvibili con i metodi allora disponibili, mentre le condizioni sperimentali risultavano proibitive - si pensi, ad esempio, alla necessità di raggiungere temperature dell’ordine di centinaia di milioni di gradi Kelvin. In tali circostanze, l’unica via praticabile era il metodo Monte Carlo. Questo approccio statistico, tuttavia, esigeva un’elaborazione numerica automatizzata: *là dove teoria ed esperimento si arrestavano, era il calcolatore a dover intervenire*. Fu l’ENIAC - dapprima nella sua configurazione originaria, poi modificato secondo i principi architettureali e di programmazione dello *stored-program* - ad essere individuato come la macchina capace di implementare i metodi probabilistici indispensabili allo studio degli ordigni nucleari.

“Più precisamente, i teorici delle armi nucleari trasformarono la nascente “macchina calcolatrice” e, nel processo, crearono *realtà alternative* alle quali sia la teoria sia l’esperimento erano legati in modo incerto”. [Galison, 1996, p.119 trad. mia enfasi mia]⁵⁴⁴

Nelle *trading zone* “basate sulla statistica, la teoria dei giochi, il campionamento e la programmazione” [Galison, 1996, p. 119, trad. mia] prendeva forma una nuova realtà artificiale, resa possibile dal metodo Monte Carlo. Queste zone di scambio erano attraversate dalla retorica sistemica della cibernetica e la riproducevano in cerchi concentrici: dalla pluralità dei saperi coinvolti alla varietà delle istituzioni partecipanti - accademiche, industriali, militari - fino ad estendersi all’intero assetto sociale. “Ma fisici e ingegneri ben presto elevarono il Monte Carlo al di sopra del modesto status di semplice schema di calcolo numerico; esso venne a costituire una realtà alternativa - in alcuni casi una realtà preferita - sulla quale si potevano condurre degli ‘esperimenti’” [Galison, 1996, p. 119, trad. mia]⁵⁴⁵.

⁵⁴² “Their common activity centered around the computer”.

⁵⁴³ “This meant that were the calculation assigned to a single computer (woman), a given thermonuclear bomb simulation would take 21 1.2 years”.

⁵⁴⁴ “More precisely, nuclear-weapons theorists transformed the nascent “calculating machine,” and in the process created alternative realities to which both theory and experiment bore uneasy ties”.

⁵⁴⁵ “During 1948 and 1949 statisticians, mathematicians, electrical engineers, and a restricted few others were called in to join with the physicists in the creation of this new trading zone in which talk of games, pseudorandom numbers, difference equations, and logical coding were common parlance. Now the style of work grew outward,

Le armi termonucleari e quelle a fissione potenziata avevano cristallizzato un nuovo modo di fare scienza. Nel corso del 1948 e del 1949 statistici, matematici, ingegneri elettrici e pochi altri selezionati furono chiamati a collaborare con i fisici nella creazione di questa nuova *trading zone*, in cui il linguaggio comune era fatto di giochi, numeri pseudocasuali, equazioni alle differenze e codifica logica. Ben presto questo stile di lavoro si espanse verso l'esterno e, attraverso una serie di conferenze, oltrepassò il dominio del segreto per entrare nella più ampia comunità dei ricercatori. Varcata quella soglia, si accese il dibattito su chi potesse essere considerato sperimentatore o teorico, e su che cosa dovesse contare come esperimento o come teoria [Galison, 1996, p.137 trad. mia]⁵⁴⁶.

Attraverso le simulazioni Monte Carlo prese forma anche la lingua franca delle *trading zone*, con Los Alamos come luogo d'origine: qui fisici, matematici, ingegneri e informatici collaboravano alla progettazione delle armi nucleari, dando vita a un linguaggio ibrido che inizialmente si presentava come un *pidgin*, ossia un mezzo provvisorio di comunicazione tra specialisti con competenze e vocabolari differenti. Con il progressivo impiego dei metodi probabilistici e computazionali, tuttavia, quel *pidgin* cominciò a strutturarsi: si formularono teoremi, si definirono criteri operativi, si fondarono riviste specializzate. Divenne un *creolo*. Negli anni Sessanta, ciò che era nato come strumento transitorio si era ormai consolidato in un linguaggio ibrido maturo, dotato di una propria autonomia operativa, che non necessitava più di traduzioni né di riferimenti a una lingua madre [Cfr. Galison, 1996, pp. 151–153].

L'ibridazione delle competenze emerse sotto la pressione delle contingenze straordinarie della Seconda guerra mondiale e della ricerca bellica, che mobilitarono risorse senza precedenti e imposero la collaborazione di saperi eterogenei. A Los Alamos, l'esperienza della simulazione intrecciava calcolo simbolico e progettazione hardware, modellazione fisica e sviluppo di ordigni, dissolvendo momentaneamente i confini disciplinari. In questo contesto operativo condiviso, la figura dei simulatori Monte Carlo riuniva in sé quelle del fisico teorico, del matematico, dell'ingegnere e del programmatore coinvolti nell'implementazione del

and through a series of conferences passed beyond the domain of the secret and into the wider community of researchers”.

⁵⁴⁶ “Thermonuclear and enhanced-fission weaponry had crystallized a new mode of doing scientific work. During 1948 and 1949 statisticians, mathematicians, electrical engineers, and a restricted few others were called in to join with the physicists in the creation of this new trading zone in which talk of games, pseudorandom numbers, difference equations, and logical coding were common parlance. Now the style of work grew outward, and through a series of conferences passed beyond the domain of the secret and into the wider community of researchers. As it crossed that fence, debate grew about who might count as an experimenter or as a theoretician, and what would count as an experiment or theory”.

metodo. Monte Carlo stesso esigeva infatti un insieme di attività - dalla generazione di numeri casuali alla valutazione delle distribuzioni, dalla scrittura di codici efficienti all'interpretazione dei risultati - che nessuna singola disciplina poteva padroneggiare in modo autonomo.

Persino a Princeton, dove la figura di von Neumann godeva di rispetto unanime e trasversale, le nuove figure ibride che andavano popolandolo la *trading zone* del calcolatore erano guardate con sospetto, se non con ostilità. Tra i verdi prati del New Jersey dove sorgeva la sede dello IAS, “gli eruditi dei piani superiori schivavano gli ingegneri come la peste” e “gli uomini del calcolatore erano visti dagli umanisti come materialisti e dai matematici come ingegneri” [Dyson, 2012, pp. 138–139, trad. it.].

Chi operava al computer, tuttavia, con la necessità di sviluppare il Monte Carlo. “preso tra una vita da macchina e una vita da simbolo [...] divenne al tempo stesso un paria e un intermediario insostituibile” [Galison, 1996, p.139 trad. mia]⁵⁴⁷, poiché, unico tra tutti, possedeva le competenze transizionali necessarie alla nuova realtà della simulazione creata dai metodi probabilistici implementati dal calcolatore. I programmatori “parlavano una lingua intermedia, una sorta di creolo formalizzato: il linguaggio delle simulazioni al computer, comprensibile tanto dai teorici quanto dagli sperimentatori” [Galison, 1996, p.155 trad. mia]⁵⁴⁸

Le simulazioni al computer - introdotte dal metodo Monte Carlo, “applicato da sessant’anni a una gamma di problemi sempre più ampia, in campi che vanno dalla fisica alla biologia fino alla finanza” [Dyson, 2012, p. 228, trad. it] - hanno assunto un ruolo sempre più centrale con l’aumento della potenza computazionale, reso possibile dai progressi dell’elettronica e dall’introduzione di nuovi strumenti di interfaccia e visualizzazione. Metropolis ricorda, nella sua ricostruzione del metodo Monte Carlo del 1987: “Quando furono introdotti i dispositivi di visualizzazione, la tentazione di sperimentare divenne quasi irresistibile, almeno per quei pochi fortunati che godevano del lusso di una politica *hands-on* (di uso diretto)” [Metropolis, 1987, pp.129-130 trad. mia]⁵⁴⁹.

La simulazione, inizialmente concepita come semplice strumento numerico, si affermò progressivamente come una rappresentazione digitale - spesso grafica - di eventi reali o

⁵⁴⁷ “Caught between a machine life and a symbol life, the computer programmer in physics became both a pariah and an irreplaceable intermediary”.

⁵⁴⁸ “This left the engineers and their successors, the computer programmers, in a peculiar position. They spoke an intermediate language, a kind of formalized creole: the language of computer simulations understood both by theorists and by experimenters”.

⁵⁴⁹ “The role of the Monte Carlo method in reinforcing such mathematics seems self-evident. When display units were introduced, the temptation to experiment became almost irresistible, at least for the fortunate few who enjoyed the luxury of a hands-on policy”.

ipotetici, la cui evoluzione poteva essere osservata nel tempo come in una sorta di realtà virtuale. Questo nuovo approccio non solo si diffuse in fisica, ma trovò applicazione crescente in ambiti quali la biologia, l'economia, la meteorologia e le scienze sociali. La centralità assunta dalle simulazioni sollevava tuttavia interrogativi sul loro statuto epistemologico: esse non potevano essere valutate secondo i parametri classici della scientificità, fondati sulla deduzione teorica o sulla riproducibilità sperimentale in senso stretto.

Con l'affermarsi della simulazione come pratica epistemica, ai parametri epistemologici classici - come la verificabilità o falsificabilità, la riproducibilità sperimentale e la pubblicità dei risultati - se ne affiancarono altri, legati alla natura ibrida e all'implementazione computazionale della simulazione. Ora erano i programmi stessi a dover soddisfare una serie di requisiti, per quanto spesso problematici: dovevano essere *aperti, portabili e modulari*.

L'*apertura dei programmi*, intesa come possibilità di renderli accessibili alla comunità scientifica, si rivelò presto problematica a causa della loro complessità crescente, che ne impediva la diffusione attraverso i mezzi, ad esempio quelli editoriali, tradizionali.

La *portabilità*, ossia l'uso di linguaggi standardizzati e la possibilità di trasferire i dati su supporti fisici tra contesti diversi; ma anche questa pratica incontrava ostacoli, sia tecnici, legati alle differenze tra macchine e ambienti operativi, sia giuridici, come le questioni relative alla proprietà dei codici e alla tutela dell'accesso. Che differenza c'è tra un campione di cellule e un set di dati?

Infine, si faceva strada l'esigenza della *modularità*, ovvero la costruzione del software secondo unità funzionali riutilizzabili - analoghe agli strumenti matematici standard nella fisica teorica - che permettessero di integrare facilmente modelli diversi in strutture computazionali complesse. [Cfr. Galison, 1996, pp.140-141]. "Ciascuna di queste risposte programmatiche - la promozione della pubblicazione, della portabilità e della modularità - fu parziale; nessuna riuscì davvero a universalizzare un insieme di pratiche che portavano l'impronta profonda della loro origine localizzata" [Galison, 1996, p.141 trad. mia]⁵⁵⁰

Sebbene non prive di criticità, tali esigenze contribuirono progressivamente a ridefinire i criteri della scientificità di cui il computer si faceva vettore e delle "realtà artificiali" da esso generate [Galison, 1996, p. 142], delineando una vera e propria *epistemologia della simulazione*. Allo stesso tempo, plasmarono in modo decisivo le forme assunte dall'hardware (ad esempio

⁵⁵⁰ "Each of these programmatic responses-advocacy of publication, portability, and modularity-was partial; none could truly universalize a set of practices that bore the deep stamp of its localized creation".

nello sviluppo grafico), dal software e dalle pratiche di programmazione, divenendo parte integrante della logica stessa che accompagnò la diffusione dei calcolatori.

A questo mutamento epistemologico se ne accompagnava uno di natura *ontologica*. Nella simulazione, infatti, non si trattava più soltanto di rappresentare fenomeni complessi, ma di *generarli* in uno spazio controllato e artificiale. Nel 1955 Herman Kahn (1922–1983) - fisico, consulente militare e futurologo della RAND Corporation (*think tank* strategico legato alla difesa statunitense)⁵⁵¹ - osservava: “In un problema Monte Carlo lo sperimentatore ha il completo controllo della propria procedura di campionamento. Se, per esempio, volesse un maiale con gli occhi verdi, i capelli ricci e sei dita, e se tale evento avesse una probabilità diversa da zero, allora lo sperimentatore Monte Carlo, a differenza dell’agricoltore, potrebbe produrre immediatamente l’animale” [Kahn, 1955, trad. mia]⁵⁵².

Con la diffusione del calcolatore, la simulazione si incardina progressivamente sui cardini delle *trading zones* e della *lingua franca* ibrida che esse comportano, per poi diffondersi in ambiti sempre più ampi. Chi padroneggia gli strumenti della simulazione non è più soltanto un matematico che risolve equazioni, né uno sperimentatore che raccoglie dati. Piuttosto, è un architetto di mondi alternativi, un costruttore di realtà artificiali che, proprio in quanto tali, possono rendere visibili aspetti altrimenti inaccessibili della realtà effettiva.

Nei problemi di fisica dei neutroni coinvolti nella bomba, persino il “massiccio supporto cibernetico” dell’ENIAC non bastava [Galison, 1996, p.126 trad.mia]⁵⁵³. Serviva un metodo nuovo, il Monte Carlo, e delle garanzie intorno alla scientificità delle simulazioni che produceva [Cfr. Galison, 2016, pp. 142-151].

Ma è necessario un passo ulteriore. Scrive Galison, che accanto alle considerazioni di natura epistemica sullo statuto della rappresentazione simulata e della sua scientificità, bisogna affiancarne un'altra. Si deve andare:

Verso una *giustificazione metafisica* della validità del Monte Carlo come forma di indagine nella filosofia naturale. L’argomento, così come fu presentato da varie persone (incluso, talvolta, lo stesso Ulam), si basava su una presunta affinità fondamentale tra il Monte Carlo e i fondamenti statistici del

⁵⁵¹ Kahn è un altro, insieme a Teller, possibili ispiratori della figura del Dottor Stranamore del film capolavoro di Stanley Kubrick del 1964.

⁵⁵² “In a Monte Carlo problem the experimenter has complete control of his sampling procedure. If for example he wanted a green-eyed pig with curly hair and six toes and if this event had a non zero probability, then the Monte Carlo experimenter, unlike the agriculturist, could immediately produce the animal”.

⁵⁵³ “But even with this massive cybernetic assistance, it was apparent that the Super problem far exceeded the ENIAC’s capacity to model, and Ulam began to search for a more efficient way to exploit the capabilities of the new machine”.

mondo stesso. In altre parole, poiché sia il metodo Monte Carlo sia la natura erano stocastici, il metodo poteva offrire uno scorcio di realtà in precedenza nascosto al matematico di orientamento analitico [Galison 1996, p.144 trad. mia]⁵⁵⁴.

La caccia all'equazione differenziale - avviata con i tentativi di integrazione meccanica di Lord Kelvin [Cfr. Goldstine, 1972, p. 43], proseguita nelle macchine analogiche di Bush, nella cibernetica wieneriana e successivamente rimodellata, in funzione delle esigenze del calcolo numerico, dagli studi di idrodinamica e da von Neumann [Cfr. Galison, 1996, pp. 122-125] -, una volta applicata ai modelli probabilistici necessari per lo studio delle reazioni neutroniche, trovava nel calcolatore la propria realizzazione: una realtà generata che conferiva concretezza a un metodo al tempo stesso rappresentazione fedele e strumento privilegiato per l'indagine della natura stocastica del mondo. L'approfondimento della metafora cibernetica avrebbe portato l'informazione e la teoria probabilistica sino nel cuore dell'hardware del calcolatore che aveva dischiuso questo mondo. "Se prima l'equazione differenziale alle derivate parziali appariva come l'arredo sublime del cielo platonico, ora i metodi Monte Carlo sembrano rappresentare davvero la struttura profondamente acausale del mondo" [Galison, 1996, p.157 trad. mia]⁵⁵⁵

Metropolis e Ulam, nell'articolo con cui rendevano pubblico il metodo Monte Carlo, concludevano la presentazione descrivendo uno spazio: "Lo 'spazio' in cui ha luogo il nostro processo è l'insieme di tutte le possibili catene di eventi, o grafi ramificati infiniti" [Metropolis, Ulam, 1949 trad. mia]⁵⁵⁶. La simulazione - computerizzata, ibrida, probabilistica - nasceva per esplorare questa dimensione. Il mondo della simulazione produceva una "liberazione dallo spazio" e una "liberazione dal tempo" [Cfr. Galison, 1996, p. 141], inizialmente sul piano scientifico e, in seguito, in tutti gli ambiti da essa toccati. Non si può non tornare al problema posto da Wiener: "L'informazione è informazione, non materia o energia" [Wiener, 1948, p. 177,

⁵⁵⁴ "All the forms of assimilation, of Monte Carlos to experimentation, that I have presented so far (stability, error tracking, variance reduction, replicability, and. so on) have been fundamentally epistemic. That is, they are all means and practices by which the researchers can argue toward the validity and robustness of their conclusions. Now I want to turn in a different direction, toward what amounted to a *metaphysical case* for the validity of Monte Carlo as a form of natural philosophical inquiry. The argument, as it was presented by a variety of people (including occasionally Ulam himself), was based on a purportedly fundamental affinity between the Monte Carlo and the statistical underpinnings of the world itself. In other words, because both Monte Carlo and nature were stochastic, the method could offer a glimpse of reality previously hidden to the analytically minded mathematician".

⁵⁵⁵ "Where the partial differential equation had appeared as the exalted furniture decorating Plato's heaven, now Monte Carlo methods appeared to re-present truly the deeply acausal structure of the world".

⁵⁵⁶ The "space" in which our process takes place is the collection of all possible chains of events, or infinite branching graphs.

trad. it.]. Analogamente - e indipendentemente da materia ed energia - essa è governata da distribuzioni stocastiche, indifferenti alla natura del mittente, del ricevente e del messaggio formalizzate dalle equazioni di Shannon.

Diversamente da una considerazione puramente epistemica del metodo, ciò che si afferma è che Monte Carlo si fondava sulla stessa ontologia probabilistica che governa il mondo fisico. Non ci si limita a dire che Monte Carlo *funziona*, ma che *è giusto, appropriato* al mondo. Così il programmatore, l'architetto e l'ingegnere della simulazione. può intervenire sulla realtà, fisica e digitale interconnesse.

Gli scrittori di narrativa hanno riconosciuto (più chiaramente dei filosofi o degli storici) le potenti e problematiche tentazioni di questa realtà artificiale. Nella straordinaria trilogia di William Gibson [la *Trilogia dello Sprawl* (1984-1988)], il *cyberspazio* - la realtà guidata dai computer, non collocata nello spazio fisico - possiede un significato pari a quello del nostro spazio tridimensionale quotidiano. Il fascino seducente del cyberspazio, inteso come alternativa per affrontare gli spigoli taglienti della vita quotidiana, è stato - ed è - evidente a coloro che lavorano con le simulazioni [Galison, 1996, trad. mia enfasi mia]⁵⁵⁷.

Il mondo dell'informazione nasce da questa liberazione dallo spazio e dal tempo, ed è proprio ciò che i protagonisti "controculturali" incontrati nelle pagine precedenti vollero esplorare attraverso i computer personalizzati - strumenti di emancipazione individuale resi possibili da una serie di iniziative imprenditoriali e animati dalla teoria unificata dell'informazione e dalla retorica della cibernetica. È lo stesso mondo immateriale, privo di coercizioni, vincoli e leggi, immaginato dai libertari del web e condensato nella *Declaration of the Independence of Cyberspace*, scritta da John Perry Barlow e pubblicata su Internet l'8 febbraio 1996.

"L'universo digitale e la bomba all'idrogeno videro la luce nello stesso momento" [Dyson, 2012, p. 250, trad. it.]. Nel loro intreccio originario, "bomba e computer" - o, potremmo dire,

⁵⁵⁷ "Fiction writers have recognized (more clearly than philosophers or historians) the powerful and problematic temptations of this artificial reality. In William Gibson's remarkable trilogy cyberspace - the unlocated computer-driven reality outside physical existence - contains as much significance as our daily three-dimensional space. The seductiveness of cyberspace as an alternative to coping with the harsh edges of the everyday is, and was, apparent to those who work with simulations".

probabilità e computazione, informazione e digitale - hanno modellato la simulazione e, con essa, gli ambiti che essa andava progressivamente a toccare.

Il medium che genera e permette di accedere a questa realtà è il calcolatore digitale. “Il computer iniziò come uno strumento-un oggetto per la manipolazione di macchine, oggetti ed equazioni. Ma, poco a poco (byte dopo byte), i progettisti di computer decostruirono la nozione stessa di strumento, poiché il computer venne a rappresentare non uno strumento, ma la natura stessa” [Galison, 1996, pp.156-157 trad. mia]⁵⁵⁸.

Monte Carlo è stato lo snodo che ha permesso questa svolta prima epistemologica, poi ontologica (con le evidenti ricadute nell’interazione dell’essere umano con la macchina e con l’informazione che questa crea ed elabora). George Dyson fa notare, il 2012, come:

Il metodo Monte Carlo è in grado di trovare soluzioni concrete per problemi che diversamente risulterebbero intrattabili, poiché la ricerca più efficiente in un ambiente inesplorato assume la forma di un cammino casuale. Gli odierni motori di ricerca, diretti discendenti dell’epoca dell’ENIAC, conservano ancora la traccia originaria del Monte Carlo: percorsi di ricerca casuale sono statisticamente ritenuti capaci di produrre risultati sempre più precisi. La genialità del metodo Monte Carlo, e dei motori di ricerca suoi “figli”, sta proprio nella capacità di ricavare soluzioni significative, a fronte di una mole immensa di informazioni, riconoscendo che il significato non sta tanto nei dati finali quanto nel percorso compiuto per arrivarci. [Dyson, 2012, p.229 trad.it.].

Una volta aperto questo universo, prima di andarne a indagare i suoi elementi costitutivi, dai e informazioni, è opportuno andare a indagare gli sviluppi tecnologici che hanno portato alla computazione personalizzata.

2. Liberi dalla paura, liberi dalla matematica: *mainframes* e programmazione

Abbandonata la Moore School il 31 marzo 1946, Presper Eckert e John Mauchly, come detto, fondarono la propria compagnia privata di calcolatori: la *Eckert-Mauchly Computer Corporation* (EMCC). All’epoca, Howard Aiken, padre del calcolatore di Harvard, dichiarava: “Il

⁵⁵⁸ “In baldest possible form: the computer began as a “tool”-an object for the manipulation of machines, objects, and equations. But bit by bit (byte by byte), computer designers deconstructed the notion of a tool itself as the computer came to stand not for a tool, but for nature itself”.

mercato commerciale non si sarebbe mai sviluppato; negli Stati Uniti c'era bisogno, forse, di cinque o sei macchine di quel tipo, ma non di più" [Ceruzzi, 2003, p. 23, trad. mia]⁵⁵⁹.

La compagnia, sotto-organizzata e con capitali insufficienti, vendeva i suoi computer a un prezzo di gran lunga inferiore rispetto al reale costo di produzione, il che portò inevitabilmente a una serie di crisi finanziarie [Haigh et al., 2016, pp. 108 trad. mia]⁵⁶⁰. Il primo computer realizzato fu il BINAC (*BINary Automatic Computer*) per la Northrop Aviation, ma fu scarsamente usato.

Tuttavia, l'impresa dei due ex Moore School, acquistata nel 1950 dalla Remington Rand e trasformata in una divisione della stessa - con un team di una dozzina di ingegneri, nella modesta sede assegnata al 3747 di Ridge Avenue, Philadelphia [Haigh, Ceruzzi, 2021, p. 23] - riuscì a concepire e produrre un calcolatore che usciva dai laboratori militari e diventava un oggetto commerciale. Questa macchina era l'UNIVAC. "L'acronimo derivava da *Universal Automatic Computer*, un nome scelto con cura. *Universal* implicava che la macchina potesse risolvere problemi affrontati da scienziati, ingegneri e aziende" [Ceruzzi, 2003, p.15, trad. mia]⁵⁶¹. L'UNIVAC fu consegnato al primo cliente, l'Ufficio del Censimento degli Stati Uniti, il 31 marzo 1951, e il suo costo finale fu stimato intorno al milione di dollari.

La macchina [fig. 22], enorme, era composta da circa 42 pannelli, occupava oltre 35 metri quadrati e pesava quasi 8 tonnellate. Utilizzava 4 bit per codificare ciascuna cifra decimale e, nel suo processore centrale, disponeva di quattro accumulatori *general-purpose* per eseguire le operazioni aritmetiche. Ogni *word* era lunga 45 bit e poteva rappresentare 11 cifre decimali più un segno, oppure sei caratteri alfabetici. L'orologio del UNIVAC operava a 2,25 MHz e permetteva di eseguire circa 465 moltiplicazioni al secondo, prestazioni in linea con quelle dell'ENIAC, ma rese più efficienti grazie alla maggiore capacità di memoria e al sistema a nastro magnetico.

Fig. 22 l'UNIVAC della *Eckert-Mauchly Computer Corporation* e la sua console

⁵⁵⁹ "In Aiken's view, a commercial market would never develop; in the United States there was a need for perhaps for five or six such machines, but no more."

⁵⁶⁰ "Eckert-Mauchly Computer Corporation, the world's first computer start-up. Understaffed and undercapitalized, it priced its computers far below the eventual cost of production, which led inevitably to a series of financial crises."

⁵⁶¹ The acronym came from "Universal Automatic Computer," a name that they chose carefully. "Universal" implied that it could solve problems encountered by scientists, engineers, and businesses".



L'imprinting del lavoro di Mauchly e Presper Eckert si riconosceva anche nella tecnologia elettronica impiegata: la *delay-line* al mercurio [Cfr. Eckert, 1980, p.531]. La memoria primaria era costituita da 100 canali, ognuno composto da registri a linea di ritardo da 10 parole, per un totale di 1.000 parole, memorizzate come impulsi acustici nei tubi al mercurio. Il processore centrale conteneva oltre 6.100 tubi a vuoto, montati in armadi disposti in un rettangolo di circa 13 metri quadrati. All'interno di questa struttura erano collocati i serbatoi delle linee di ritardo [Cfr. Haigh, Ceruzzi, 2021, pp.23-24; Ceruzzi, 2003, pp.28-29]. "Molte delle sue caratteristiche innovative sarebbero poi diventate comuni, tra cui l'elaborazione alfanumerica oltre a quella numerica, i nastri magnetici per la memoria di massa e circuiti chiamati *buffer*, utilizzati per evitare che il processore restasse bloccato in attesa del trasferimento dei dati sul nastro" [Haigh, Ceruzzi, 2021, p.24 trad. mia]⁵⁶².

UNIVAC, che, in linea con le altre macchine statunitensi che abbiamo analizzato, "rifletteva il modo in cui gli esseri umani, utilizzando calcolatrici meccaniche, tavole numeriche e carta e matita, eseguivano i calcoli scientifici" [Ceruzzi, 2003, p.15 trad. mia]⁵⁶³. Non impiegava tuttavia le schede perforate, bensì usava un nastro.

UNIVAC era, nell'immaginario pubblico – una sua versione evoluta, l'UNIVAC #5, in una diretta televisiva predisse la vittoria alle presidenziali del 1952 di Eisenhower [Cfr. Haigh,

⁵⁶² "Many of its novel features later became commonplace, among them alphanumeric as well as numeric processing, magnetic tapes for bulk memory, and circuits called buffers to stop the processor being tied up waiting for data to transfer to tape".

⁵⁶³ "That is, the flow of instructions and data in the UNIVAC mirrored the way humans using mechanical calculators, books of tables, and pencil and paper performed scientific calculations".

Ceruzzi, 2021, p.27] - un *cervello automatico* che sapeva dove trovare i dati che processava e come elaborarli:

Per la maggior parte dei clienti, ciò che rendeva rivoluzionario l'UNIVAC non era tanto il design a programma memorizzato né il processore elettronico, quanto l'uso del nastro al posto delle schede perforate. Per loro, il carattere "automatico" della macchina risiedeva nella sua capacità di scorrere un rullo di nastro, trovare il record corretto o un insieme di record, eseguire un'elaborazione su di essi e restituire i risultati nuovamente su nastro. In un'installazione a schede perforate, queste operazioni venivano svolte da operatori umani, che dovevano trasportare grandi mazzi di schede da una macchina all'altra. [Ceruzzi, 2003, p.30]⁵⁶⁴

Il primo UNIVAC del 1951 fu impiegato dall'Ufficio del Censimento per elaborare su nastro magnetico i dati di 11 milioni di schede perforate in cui erano contenuti i dati dei cittadini. Nel 1952, l'UNIVAC #2 fu installato al Pentagono per il Progetto SCOOP (*Scientific Computation of Optimum Problems*), nato durante la guerra per ottimizzare la logistica dei rifornimenti militari e successivamente applicato alla logistica delle basi dell'Air Force dislocate in tutto il globo. Il progetto contribuì allo sviluppo della *programmazione lineare* –un metodo matematico per ottenere il miglior risultato quando vincoli e obiettivo sono espressi da relazioni lineari.

Il modello acquistato dalla General Electric per automatizzare le operazioni contabili, logistiche e di inventario, il 15 ottobre 1954 generò per la prima volta le buste paga per lo stabilimento di Appliance Park a Louisville. [Cfr. Haigh, Ceruzzi, 2021, pp.26-27].

Un'Utopia igited- o una distopia, secondo i contestatori contro-culturali che abbiamo incontrato - in cui i giganteschi cervelli elettronici razionalizzavano la società e automatizzavano lavoro e produzione, sembrava realizzarsi attraverso il computer commerciale degli anni '50. [Cfr. Haigh, Ceruzzi 2021, p.28].

La risposta della concorrente della Remington Rand non si fece attendere. Adottando i principi architettonici dell'EDVAC e avvalendosi della consulenza di John von Neumann, l'IBM lanciò nel maggio del 1952 il modello 701 (tra i responsabili di alto livello del progetto vi era anche l'ingegnere Nat Rochester, che sarebbe poi stato uno dei promotori del celebre

⁵⁶⁴ "For most customers, what was revolutionary about the UNIVAC was not so much its stored-program design or even its electronic processor. It was the use of tape in place of punched cards. To them, the "Automatic" nature of the machine lay in its ability to scan through a reel of tape, find the correct record or set of records, perform some process in it, and return the results again to tape. In a punched card installation, these tasks were performed by people who had to".

Dartmouth Summer Research Project del '56, durante il quale fu coniato il termine *Artificial Intelligence*.).

Il modello fu commercializzato con il nome di "*Defense Calculator*", rivolto principalmente agli enti militari del governo statunitense, a cui la macchina veniva affittata per 15.000 dollari al mese. Il 701, anch'esso di dimensioni imponenti (pesava circa 10 tonnellate), era un calcolatore *general purpose* a programma memorizzato "binario parallelo progettato principalmente per risolvere grandi problemi nel calcolo scientifico e ingegneristico [...] È il primo sistema di questo tipo a essere prodotto in serie. Il computer è controllato da un programma memorizzato composto da istruzioni a indirizzo singolo" [Buchholz, 1953]⁵⁶⁵.

Il 701 impiegava 72 tubi Williams per la memoria principale *high-speed*, ognuno dei quali conteneva 1.024 bit. L'insieme di questi tubi catodici consentiva di memorizzare 2.048 parole da 36 bit ciascuna. L'accesso alla memoria era particolarmente rapido, poiché la macchina poteva trasferire una parola alla volta leggendo simultaneamente un bit da ciascuno dei 36 tubi attivi, a differenza del calcolatore Remington basato su linee di ritardo, che restituiva i bit in sequenza.

Diversamente dell'UNIVAC, nel 701 il processore centrale gestiva direttamente le operazioni di input/output, costringendo tutti i trasferimenti di dati a passare attraverso un unico registro, con conseguente perdita di velocità. Tuttavia, questa era in parte compensata dall'impiego di nastri in plastica, che garantivano prestazioni superiori rispetto ai nastri metallici del sistema Remington. La macchina era dotata per i sistemi di input-output di una stampante [*printer*] di lettori di schede perforate e di nastro magnetico, tutte sotto il controllo dell'unità centrale [Cfr. Buchholz, 1953; Haigh, Ceruzzi, 2021, pp.29-30; Ceruzzi, 2003, p.34-36]⁵⁶⁶.

L'IBM 701 inizialmente fu impiegato per compiti militari come il design di armi, il calcolo di traiettorie balistiche o aeree spaziali, l'attività computazionale legata allo sviluppo di ordigni atomici (un modello fu consegnato a Los Alamos all'inizio del 1953), ma ben presto fu impiegato dagli organi di difesa del governo per mansioni simili a quelle di UNIVAC - ebbene non per compiti civili - come logistica, contabilità report finanziari. E anch'esso prevede una vittoria di Eisenhower, la sua rielezione del 1956 [Cfr. Haigh, Ceruzzi, 2021, p.30].

⁵⁶⁵ "The 701 is a parallel binary computer designed primarily to solve large problems in scientific and engineering computation. It is the first such system to be manufactured in quantity. The computer is controlled by a stored program of single-address instructions".

⁵⁶⁶ Per una descrizione completa e accessibile delle caratteristiche tecniche e del design del modello 701 si rimanda a Buchholz, 1953.

I calcolatori commerciali presi in esame - e i loro successori - furono anche il terreno in cui prese forma un'ulteriore innovazione: quella relativa alla *programmazione*. Nelle macchine analizzate nelle pagine precedenti si è più volte sottolineato come l'impostazione della macchina per l'esecuzione di un compito specifico richiedesse l'intelligenza, la creatività e il lavoro minuzioso dell'operatore (o, più spesso, dell'operatrice), impegnato nel cablaggio e nella configurazione di prese, cavi e interruttori, oppure nella preparazione di voluminosi mazzi di schede perforate o nella punzonatura di pattern di fori sul nastro destinato alla successiva elaborazione da parte della macchina.

Da un certo punto di vista, se l'*hardware* della macchina racchiudeva in potenza le capacità di calcolo, il *software* risiedeva ancora nella mente (e nelle mani) delle programmatrici. Con l'avvento delle macchine *general purpose* a programma memorizzato emerse invece una qualità latente del calcolatore, riguardante precisamente il modo in cui le istruzioni potessero essere fornite alla macchina. Si comprese come il programma memorizzato potesse essere modificato dai risultati stessi di un'operazione [Cfr. Ceruzzi, 2015, p. 58].

Le liste di istruzioni contenute in *First Draft of a Report on the EDVAC* o nel manuale per programmatori redatto da Turing per la macchina di Manchester rispondono anche a questa caratteristica⁵⁶⁷. In gran parte dei calcolatori presi in esame “bisognava prepararsi selezionando i mazzi di schede appropriati, scrivendo e perforando codici di transizione su schede, e raggruppando il risultato in un nuovo mazzo di schede” Negli anni '50 iniziarono a svilupparsi sequenze di istruzioni che consentivano alla macchina di occuparsi di questo compito. “Tali programmi vennero chiamati “compilatori”. [Haigh Ceruzzi, 2021, p.85 trad. mia]⁵⁶⁸, termine nato proprio dall'attività di compilazione delle schede .

Una figura chiave dello sviluppo dei *compilers* è stata Grace Hopper la quale, dopo l'esperienza ad Harvard con Aiken, lavorò proprio su UNIVAC, fungendo da ponte nella transizione dello sviluppo dei calcolatori e della loro programmazione dal mondo accademico-militare a quello commerciale. Nel gergo dell'informatica contemporanea un compilatore è un programma che traduce istruzioni scritte in un linguaggio comprensibile agli esseri umani in codici binari eseguibili da un computer [Aho et al., 2006].

⁵⁶⁷ Per la storia dello sviluppo dei linguaggi di programmazione di rimanda a Knuth, Pardo, 1980, pp.197-274; e. Wexelblat, 1981; Gabbrielli, Martini, 2011, 535-558.

⁵⁶⁸ If these sequences were punched onto decks of cards, a program could be prepared by selecting the appropriate decks, writing and punching transitional codes onto cards, and grouping the result on a new deck of cards [...] these programs were called “compilers.”

Grace Hopper aveva però in mente un'idea diversa quando elaborò il suo programma compilatore per l'UNIVAC. Per la futura ammiraglio della Marina degli Stati Uniti, il *compiler* era un programma destinato a gestire subroutine - relative ai compiti computazionali della macchina - archiviate in apposite librerie. La sua funzione consisteva nel prelevare il codice della subroutine e inserirlo direttamente nel punto del programma principale in cui il programmatore intendeva impiegarlo [Cfr. Hopper, 1952; Knuth, Pardo, 1980, p. 34].

Il primo compilatore nel senso moderno fu sviluppato dall'ingegnere elettronico J. H. Laning (1920–2012) e dal logico N. Zierler (1921–2004), per il calcolatore *Whirlwind* del MIT⁵⁶⁹. Questo programma, rimasto anonimo, riceveva in input i comandi digitati dall'utente e generava in output un codice macchina nuovo e autonomo, capace non solo di eseguire quei comandi, ma anche di gestire automaticamente le locazioni di memoria, i cicli ripetitivi e le altre operazioni di controllo e organizzazione interna [Cfr. Ceruzzi, 2003, pp.86-87; Knuth, Pardo, 1980, pp. 236-240].

Questo compilatore fu praticamente ignorato. John Backus lasciò intendere che la ragione risiedesse nella minaccia che esso rappresentava per la "casta sacerdotale" [*priesthood*] dei programmatori, i quali, con un atteggiamento che perdura ancora oggi, desideravano rimanere gli unici custodi della conoscenza del codice macchina, l'unico linguaggio con cui, secondo loro, si poteva davvero comunicare con il calcolatore e di cui erano i depositari [Cfr. Backus, 1980, p.128-129; Ceruzzi, 2003, p.86].

L'idea alla base del compiler ha ricadute contemporanee notevolissime. Essa "è alla base delle interfacce utente moderne, siano esse basate sull'uso del mouse per selezionare icone o elementi su uno schermo, su un touchpad, sull'input vocale-o qualsiasi altro mezzo" [Ceruzzi, 2015, p.59 trad. mia]⁵⁷⁰.

Lo sviluppo e il miglioramento ha fatto sì che questi sistemi siano stati via via conosciuti come *linguaggi di programmazione*. "L'emergere di tale termine era legato al fatto che condividevano alcuni attributi, seppur limitati, con il linguaggio naturale, come l'esistenza di regole sintattiche" [Ceruzzi, 2003, p.87 trad. mia]⁵⁷¹.

⁵⁶⁹ Su *Whirlwind* si veda Everett, 1980, pp.365-394.

⁵⁷⁰ "That concept is at the root of modern user interfaces, whether they use a mouse to select icons or items on a screen, a touch pad, voice input-anything".

⁵⁷¹ "These systems eventually were improved and came to be known as Programming Languages. The emergence of that term had to do with their sharing of a few restricted attributes with natural language, such as rules of syntax".

I linguaggi di programmazione possono essere classificati in base al tipo di *astrazione* che adottano, concetto già introdotto all'inizio del nostro discorso e centrale nella progettazione dei linguaggi stessi. Per astrazione si intende “la capacità di definire e poi utilizzare strutture o operazioni complesse in modo tale da poter ignorare molti dei dettagli” [Sebesta, 2012, p. 14, trad. mia]⁵⁷².

Finora si è parlato dei calcolatori come macchine fisiche incaricate di eseguire operazioni, ossia gli “algoritmi”. Per comprendere i linguaggi di programmazione - che rappresentano le istruzioni che la macchina deve “comprendere” per poterle eseguire - è necessario introdurre il concetto di *macchina astratta*, già incontrato nel lavoro di Turing. L'introduzione di questo concetto consente di chiarire il passaggio dal *computer umano* all'insieme delle attività programmate nel calcolatore (insieme ai limiti computazionali e di gestione della complessità).

Dato un linguaggio di programmazione scritto in L , dotato di una sintassi e di una semantica, “un programma di L . (o programma scritto in L) dunque non è altro che un insieme finito di istruzioni di L ”, “Definiamo una macchina astratta per L e la indichiamo con M_L , un qualsiasi insieme di strutture dati e di algoritmi che permettano di memorizzare ed eseguire programmi scritti in L ”. [Gabrielli, Martini, 2011, p.2] Una macchina astratta generica M_L è costituita dalla memoria e da un interprete, per immagazzinare dati e per eseguire le istruzioni fornite in un certo linguaggio [Gabrielli, Martini, 2011, p.2].

Poste queste definizioni basilari si può distinguere tra linguaggi di *alto livello* e di *basso livello* a seconda della vicinanza, nella relazione tra uomo e macchina, all'essere umano o all'hardware. I linguaggi di *basso livello* sono quelli “di macchine astratte molto vicine alla macchina o che coincidono con questa”. I linguaggi di *alto livello* sono quelli, invece, che, “mediante opportuni meccanismi di astrazione, permettono di usare costrutti che prescindono dalle caratteristiche fisiche del calcolatore. I linguaggi di alto livello sono dunque adatti ad esprimere algoritmi in modo relativamente facile per l'utente umano” [Gabrielli, Martini, 2011, p.5].

Le macchine come ENIAC, in cui il processo di codifica era pressoché assente, funzionavano secondo il *linguaggio macchina*, in cui le istruzioni elementari erano “di immediata esecuzione da parte del processore”. Se questi ultimi rappresentano la prima

⁵⁷² “Briefly, abstraction means the ability to define and then use complicated structures or operations in ways that allow many of the details to be ignored. Abstraction is a key concept in contemporary programming language design”.

generazione di linguaggi (1GL), i primi due corrispondono alla seconda (2GL) e alla terza (3GL) [Gabrielli, Martini, 2011, p.537].

Per lungo tempo continuarono ad essere impiegati linguaggi di basso livello che producevano poche istruzioni macchina per ogni istruzione data dal programmatore. “Ogni comando veniva tradotto uno a uno in un numero binario corrispondente, che il computer poteva eseguire direttamente. Un programma non veniva “compilato” ma “assemblato”, e il programma che compiva questa operazione era detto *assemblatore*” [Ceruzzi, 2003, p.87 trad. mia enfasi mia]⁵⁷³. Uno dei meriti dei calcolatori commerciali fu quello di sviluppare i linguaggi, attraverso l'introduzione del concetto di compilatore, verso quelli di alto livello. Il primo linguaggio che è possibile considerare come di alto livello è stato FORTRAN (*Formula Translation*) elaborato da John Backus per l'IBM 704, un successore del 701, nel 1957.

Il modello IBM 704 [fig. 23], successore del 701, introdusse tre miglioramenti chiave: l'aggiunta di registri indice per semplificare la programmazione; l'implementazione hardware dell'aritmetica in virgola mobile, di cui l'ENIAC era privo - un limite forse sottovalutato da von Neumann, che sembrava ritenere che tutti disponessero del suo talento matematico! - e, soprattutto, la sostituzione dei “capricciosi” tubi Williams con la memoria a nuclei magnetici (*core memory*), ideata dal sino-americano An Wang (1920–1990), già allievo di Aiken ad Harvard. In questa memoria i nuclei, piccoli anelli magnetici, erano disposti in un piano e attraversati orizzontalmente e verticalmente da fili che passavano nel foro di ciascuno. “Ogni nucleo memorizza un bit di informazione, magnetizzandosi in uno dei due possibili stati. Il bit desiderato viene selezionato facendo passare corrente attraverso una combinazione unica di fili orizzontali e verticali, e si legge da un filo di uscita condiviso” [Haigh, Ceruzzi, 2021, p.31 trad. mia]⁵⁷⁴

⁵⁷³ “Each code was translated, one-to-one, into a corresponding binary number that the computer could directly execute. A program was not compiled but “assembled,” and the program that did that was called an “assembler.”

⁵⁷⁴ “Each core stores one bit of information, by magnetizing it one of two possible ways. The desired bit is selected by passing current along a unique combination of horizontal and vertical wires and is read from a shared output wire”.

Fig.23 l'IBM 704



Il sistema a nuclei magnetici da 32768 parole del 704 [Bashe et al., 1981] presentava vantaggi concreti rispetto a quelli introdotti “in potenza” dal tubo catodico: la memoria era non volatile – i nuclei mantenevano l’informazione anche senza alimentazione; collegando più piani si potevano leggere simultaneamente i bit di un’intera parola, riducendo i tempi di latenza legati al passaggio sequenziale dei bit davanti al lettore, e permetteva lo sviluppo dell’accesso casuale e rapido all’informazione, ossia la RAM (*Random Access Memory*), in cui il tempo di accesso è indipendente dalla posizione del dato. [Cfr. Haigh, Ceruzzi, 2021, p.31-32]. Il 704 raggiungeva una velocità di circa 40.000 operazioni in virgola mobile al secondo.

Sebbene legato ad un preciso calcolatore, FORTRAN presentava elementi che potremmo definire quasi *funzionalisti*, “con molti costrutti sostanzialmente indipendenti da una macchina specifica” [Gabrielli, Martini, 2011, p.540]. Veicolato dal successo commerciale dell’IBM, FORTRAN, che è usato tutt’oggi, aveva tra i suoi punti di forza la *sintassi*, molto simile a quella dell’algebra tradizionale [Cfr. Ceruzzi, 2003, pp.90-91; Backus, 1980, pp.129-133; Knuth, Pardo, 1980, pp.241-243 e pp.259-261].

FORTRAN, progettato per applicazioni scientifiche [Grundin, 2017, p.16]⁵⁷⁵, presentava un’altra caratteristica apprezzata dagli utenti e che i linguaggi di programmazione di successo

⁵⁷⁵ “Designed for scientific computing and released in 1957”.

hanno ereditato, l'idea che il linguaggio “nascondesse i dettagli del funzionamento interno della macchina, lasciando a loro il compito di concentrarsi sulla risoluzione dei propri problemi, anziché su quelli della macchina”. Un linguaggio di programmazione efficace deve “nascondere alcuni ma non tutti gli aspetti interni della macchina al programmatore” [Ceruzzi 2003, p.91 trad. mia]⁵⁷⁶.

Queste innovazioni vanno inserito nel quadro non neutrale del ruolo che il computer andava via via assumendo. L'IBM⁵⁷⁷ aveva presentato il 701 come *Defense Calculator* pensato per una clientela militare. Nel 1952 si aggiudicò un contratto per oltre mezzo miliardo di dollari il dipartimento della difesa USA per la realizzazione del progetto SAGE (*Semi-Automatic-Ground-Environment*) pensato per rilevare la presenza di bombardieri sovietici – l'URSS aveva appena testato la sua bomba atomica e sullo sfondo c'erano gli eventi della guerra di Corea – ed elaborare traiettorie di intercettazione qualora fossero penetrati nello spazio aereo degli Stati Uniti [Astrahan, Jacobs, 1983]. Il fulcro del sistema, il cui centro era il MIT, vi era “un computer [basato su Whirlwind] incaricato di coordinare le informazioni raccolte da fonti sparse, elaborarle e presentarle in una combinazione di forme testuali e grafiche” [Haigh, Ceruzzi, 2021, p.87 trad. mia]⁵⁷⁸. L'intero centro direzionale occupava quattro piani di un edificio, con uno, il secondo, totalmente occupato dai calcolatori⁵⁷⁹.

Le forze armate avevano bisogno di un calcolatore attivo 24 ore su 24 ed affidabile. Per questo la tecnologia elettronica scelta per il progetto fu quella dei nuclei magnetici di memoria [Forrester, 1983]. Il progetto originale prevedeva 8192 *words* da 33 bit, distribuite su due banchi di 33 piani, ciascuno formato da una matrice di 64×64. Quando le esigenze applicative crebbero, fu sviluppata una nuova unità da 256×256 parole per sostituire uno dei banchi originali. L'IBM e il MIT, perfezionarono delle tecniche per la produzione industriale uniforme, affidabile e basso conto di questa memoria insieme a metodi per testarli [Astrahan, Jacobs, 1983]⁵⁸⁰.

⁵⁷⁶ “FORTRAN's initial success illustrates how readily users embraced a system that hid the details of the machine's inner workings, leaving them free to concentrate on solving their own, not the machine's, problems; The successful and long-lasting computer languages, of which there are very few, all seem to share this quality of hiding some, but not all, of a computer's inner workings from its programmers”.

⁵⁷⁷ Per una descrizione dei modelli IBM dagli antecedenti del 701, come il calcolatore elettromeccanico SSEC del 1948 sino al 7010 del 1962 si rimanda a Bashe et al., 1981.

⁵⁷⁸ “It relied on computers to process information gathered using radar, aircraft, telephone lines, radio links, and ships and present it in a combination of textual and graphical form”.

⁵⁷⁹ Per la descrizione di un centro direzionale SAGE si veda Jacobs, 1983.

⁵⁸⁰ In Astrahan, Jacobs, 1983 si fornisce un esaustivo elenco delle innovazioni tecnologiche del progetto SAGE.

A maggior tutela dell'affidabilità, dopo aver presentato un prototipo nel 1955 e completato la produzione del primo modello nel 1956, l'IBM escogitò per SAGE un sistema basato su un duplice computer in parallelo. In caso di guasto di uno dei due l'attività si spostava sull'altro calcolatore grazie ad un interruttore. Alla fine l'IBM consegnò 30 sistemi all'esercito USA [Cfr. Everett et al., 1983; Ceruzzi, 2003, p.51]. Un cambio tecnologico nella tecnologia missilistica intercontinentale sovietica rese SAGE inutile.

Tuttavia, in un altro senso importante, il sistema SAGE “funzionava”. Funzionava per la comunità scientifica, che lo utilizzò per perseguire importanti obiettivi intellettuali e tecnici. Funzionava come politica industriale, fornendo finanziamenti governativi a un settore industriale emergente. Ma, forse più importante, SAGE funzionava come ideologia, creando l'impressione di una difesa attiva che placava, in parte, il senso di impotenza generato dalla paura nucleare [Edwards, 1996, p.110 trad. mia]⁵⁸¹.

L'ultimo SAGE originale fu spento solo nel 1983 in Ontario [Jacobs, 1983]. SAGE produsse due delle innovazioni maggiori della storia della computazione, “la rete tra computer e sensori e lo sviluppo della grafica interattiva” [Haigh, Ceruzzi, 2021, p.87 trad. mia]⁵⁸².

Il sistema raccoglieva ed elaborava dati provenienti da fonti disparate come i computer, i radar, le navi, aerei e dai messaggi telefonici e radio. E questi dati dovevano essere elaborati, resi graficamente e in modo testuale in tempo reale e comunicati a tutti i sistemi interconnessi. Il calcolatore al centro dell'elaborazione era usato per “il controllo e la gestione delle informazioni più che per la computazione” [Turner, 2006, p.27 trad. mia]⁵⁸³. L'IBM creò un sistema di visualizzazione dati avanzatissimo per l'epoca, basato su una consolle di visualizzazione a tubo catodico, una tastiera e una pistola luminosa [*light gun*]. Nei display grafici, il fascio di elettroni veniva modellato tramite una maschera per assumere la forma di uno dei 64 caratteri, poi deviato nella posizione desiderata sullo schermo. I display testuali utilizzavano una maschera simile ma proiettavano l'immagine su uno schermo a memoria anziché su fosforo, mantenendola visibile senza riscrittura continua [Astrahan, Jacobs, 1983].

⁵⁸¹ “Still, in another important sense, SAGE did “work.” It worked for the research community, which used it to pursue major intellectual and technical goals. It worked as industrial policy, providing government funding for a major new industry. Perhaps most important, SAGE worked as ideology, creating an impression of active defense that assuaged some of the helplessness of nuclear fear”. Sul ruolo dello sviluppo della computazione e degli studi di cibernetica e dell'Intelligenza Artificiale nel contesto geopolitico e culturale del “mondo chiuso” della Guerra Fredda, si veda Edwards, 1996.

⁵⁸² “SAGE introduced more fundamentally new features to computing than any other project of its era, including networking computers and sensors and the development of interactive computer graphics”.

⁵⁸³ “The Whirlwind was about to become the first interactive digital computer to be used (by SAGE personnel) primarily for information management and control rather than computation”.

Questi sviluppi nell'interfaccia uomo-macchina davano pieno significato al nome scelto per il sistema: *Semi-Automatic Ground Environment*, dove l'elemento centrale era proprio il termine "Semi". Come osservano Haigh e Ceruzzi, "l'Aeronautica voleva che tutti sapessero che, anche con tutta l'acquisizione e l'elaborazione in tempo reale dei dati radar, le decisioni sarebbero spettate agli esseri umani" [Haigh, Ceruzzi, 2021, p.87 trad. mia]⁵⁸⁴.

L'integrazione della gestione dell'informazione all'interno di un sistema misto uomo-macchina rappresentava un ulteriore avanzamento - militare, politico e sociale - della metafora cibernetica dell'organizzazione, già promossa a livello intellettuale dalle *Macy Conferences* o, ancor prima, dalla *Società Teleologica*. La *trading zone* tra saperi diversi e il linguaggio comune che andava emergendo tra specialisti di discipline eterogenee, accomunati da un orientamento sistemico, richiamavano da vicino quanto già sperimentato al Rad Lab. Anche SAGE, infatti, "coinvolgeva un'ampia e complessa rete di attori militari, industriali e accademici; e organizzava il proprio lavoro in maniera altamente collaborativa" [Turner, 2006, p. 27, trad. mia]⁵⁸⁵.

La coordinazione e la gestione del sistema SAGE furono distribuite e interdisciplinari, pur rientrando nell'inquadramento gerarchico degli apparati di difesa statunitensi: la diversità delle tecnologie impiegate, la varietà dei dati raccolti, la molteplicità delle competenze necessarie e l'esigenza di produrre informazioni comprensibili e utilizzabili dal decisore umano rendevano indispensabile questa impostazione. Tutti questi elementi - incluso quello umano - venivano concepiti come parti di un "organismo", un sistema integrato di elaborazione dell'informazione destinato a proteggere la popolazione e l'intero assetto sociale dalla minaccia atomica. Se dal punto di vista pratico SAGE si rivelò inefficace, esso divenne tuttavia l'archetipo delle conseguenze "sociotecniche" derivanti dall'impiego dei nuovi sviluppi del calcolo elettronico.

Come evidenzia Fred Turner, furono gli stessi vertici militari statunitensi – in particolare il *Defense Systems Engineering Committee* dell'amministrazione Truman, lo stesso che avrebbe poi promosso l'integrazione dei calcolatori Whirlwind nel progetto SAGE – ad appropriarsi del vocabolario della cibernetica e ad approfondirne la retorica all'inizio degli anni '50. In un loro rapporto si legge:

⁵⁸⁴ "The Air Force wanted everyone to know that even with all of the real- time acquisition and processing of radar data, decision making would rest with humans".

⁵⁸⁵ "Like the Rad Lab before it, SAGE was based at MIT; involved a complex range of military, industrial, and academic players; and organized its work in a highly collaborative fashion".

Il sistema di difesa aerea è un organismo [...] Ma cosa sono, dunque, gli organismi? Sono di tre tipi: organismi animati, che comprendono gli animali e i gruppi di animali, compresi gli esseri umani; organismi parzialmente animati, che coinvolgono animali insieme a dispositivi inanimati come nel sistema di difesa aerea; e organismi inanimati, come i distributori automatici. Tutti questi organismi hanno in comune: componenti sensoriali, mezzi di comunicazione, dispositivi di analisi dei dati, centri di giudizio, direttori dell'azione e agenti effettori, ovvero esecutori [...] È funzione di un organismo [...] raggiungere uno scopo definito [Air Defense System: ADSEC FINAL Report, 24 ottobre 1950, in Turner, 2006, pp.27-28 trad.mia]⁵⁸⁶.

Il sistema SAGE offrì una prova “del tipo di innovazione che può essere ottenuto quando il costo è secondario rispetto alle prestazioni. Questo tipo di ambiente è difficile da creare in un'azienda orientata al profitto commerciale, ma SAGE fornì l'ambiente adeguato” [Astrahan, Jacobs, 1983, p. 347 trad. mia]⁵⁸⁷.

In questo ambiente, centinaia di programmatori inesperti ricevettero la loro formazione teorica e pratica [Jacobs, 1983]. Questi poi si riversarono nei nascenti dipartimenti di computer science, nell'industria e nei progetti militari e civili che nascevano e si sviluppavano intorno alle innovazioni della computazione [Cfr. Turner, 2006, p.28]. Questo piccolo esercito di nuovi esperti del computer si riversò, a tutti i livelli, nella società statunitense, diffondendo non solo le innovazioni tecnologiche del progetto, ma anche la retorica che lo ispirava

L'IBM, archetipo dell'“azienda orientata allo sviluppo commerciale”, non solo trasse vantaggio economico dal progetto SAGE, ma fece propri anche i progressi tecnologici da esso derivati. Tra questi vi erano le innovazioni hardware, come la memoria a nucleo magnetico [*core memory*] più affidabile e meno costosa, poi impiegata nel modello 704 [Jacobs, 1983], e il patrimonio di competenze sviluppato intorno alle innovazioni software coinvolte nel progetto, come lo *shared data pool* accessibile a diversi sottosistemi, che divenne centrale nella gestione dei *data base* nella computazione aziendale [Cfr. Haigh, Ceruzzi, 2021, p. 89].

⁵⁸⁶ “is an organism... What then are organisms? They are of three kinds: animate organisms, which comprise animals and groups of animals, including men; partly animate organisms which involve animals together with inanimate devices such as in the Air Defense System; and inanimate organisms such as vending machines. All these organisms possess in common: sensory components, communication facilities, data analyzing devices, centers of judgment, directors of action, and effectors, or executing agencies... It is the function of an organism... to achieve some defined purpose”.

⁵⁸⁷ “The SAGE system provides a demonstration of the kind of innovation that can be achieved when cost is secondary to performance. This kind of environment is difficult to create in a commercially oriented company, but SAGE provided the environment”.

A ciò si aggiungono le innovazioni nel campo dell'interfaccia e nelle basi dell'interconnessione tra calcolatori: molti dei programmatori formati su SAGE collaborarono alla nascita di ARPANET, l'antesignana di Internet [Cfr. Turner, 2006, p. 28].

La *forma mentis* acquisita dal personale nel progetto SAGE – e la metafora cibernetica a più livelli che esso impiegava – si sarebbe riverberata, grazie alla supremazia industriale dell'IBM, in tutti gli ambiti che i suoi, e della concorrenza, *Giant Brains* andavano plasmando: la scienza, l'economia, la politica e la società nel suo insieme. Fu a questa visione che si ribellarono i contestatori incontrati all'inizio di questo lavoro, ed è qui che si innestano le dinamiche scientifiche, economiche, sociali, culturali e contro-culturali che portarono alla forma personalizzata – *small scale* - del computer [Cfr. Turner, 2006, p. 28; Ceruzzi, 2003, p. 53].

Questi sviluppi si intrecciarono con le dinamiche di miglioramento tecnologico, sia hardware che software, dei calcolatori. Il team UNIVAC raccolse la sfida posta dal successo di FORTRAN sviluppando, nel 1959, un proprio linguaggio di alto livello: il COBOL (*Common Business Oriented Language*). All'interno di questo stesso gruppo lavorava Betty Holberton (1917–2001), già una delle sei programmatrici dell'ENIAC, a cui Mauchly affidò lo sviluppo delle routine di ordinamento per l'UNIVAC - un compito cruciale, su cui aveva riflettuto anche von Neumann, che nel 1945 aveva fornito una lista di istruzioni per l'ordinamento su EDVAC, considerandolo essenziale per l'universalità della macchina. L'ordinamento, infatti, era necessario persino per una semplice modifica a un conto cliente: occorreva localizzare il record sul nastro, aggiornarlo e riscrivere l'intero file in ordine, con conseguente movimentazione di grandi quantità di dati [Cfr. Haigh, Ceruzzi, 2021, p. 65; Ceruzzi, 2003, p. 90].

Parte del successo di COBOL si deve alla richiesta del Dipartimento della Difesa del governo USA, che nel 1959 istituì un comitato per lo sviluppo di un linguaggio comune per le applicazioni aziendali dei calcolatori; l'anno seguente, il governo statunitense dichiarò che non avrebbe acquistato macchine che non supportassero COBOL. Con questo linguaggio - uno dei primi standardizzati - il codice si slegava da una macchina specifica, "al punto che uno stesso programma poteva essere eseguito su computer diversi di diversi produttori con gli stessi risultati" [Ceruzzi, 2003, p. 92, trad. mia]⁵⁸⁸. L'universalità - computazionale e di diffusione della macchina - era legata anche alla possibilità di disporre di interfacce comuni, cioè sistemi di programmazione e altri dispositivi di mediazione con l'essere umano: "ciò che era necessario,

⁵⁸⁸ "As a result, COBOL became one of the first languages to be standardized to a point where the same program could run on different computers from different vendors and produce the same results". Nel dicembre 1960 il programma per UNIVAC fu fatto girare su un calcolatore RCA.

per andare oltre, era un'interfaccia che potesse essere utilizzata per molti scopi diversi. Il calcolatore digitale 'universale' aveva bisogno di un'interfaccia 'universale'" [Boden, 2006, p. 1074]⁵⁸⁹.

COBOL è figlio del lavoro di Grace Hopper sui compilatori per UNIVAC, in particolare il B-0 (noto anche come FLOW-MATIC o MATH-MATIC), orientato alle applicazioni aziendali. Hopper "descriveva il suo obiettivo come quello di liberare i matematici affinché potessero dedicarsi alla matematica". Il suo obiettivo, generalizzato, è alla base del tentativo dell'HCI [*Human-Computer Interaction*] di "liberare gli utenti affinché potessero concentrarsi sul proprio lavoro. "[Grundin, 2017, p. 16 trad. mia]⁵⁹⁰.

La concezione della relazione uomo-macchina che ispirava Grace Hopper è emblematica della traiettoria che si stava delineando: le operazioni cognitive e procedurali venivano progressivamente astratte nella relazione con il calcolatore, mostrando le potenzialità e i limiti di quest'ultimo come sostituto di alcune facoltà dell'essere umano. Un'operazione "vecchissima" come la soluzione di un problema matematico, afferma Hopper, al pari di quella della produzione "di un'automobile, di una rotaia, di una scatola di pomodori", rispecchia le fasi di "input di dati matematici, controllo da parte del matematico, fornito di memoria, formule, tavole, matita e carta; il cervello svolge il calcolo e produce risultati". Macchine come UNIVAC promettono di "sostituire, per quanto possibile, il cervello umano con un calcolatore digitale elettronico". Con il lavoro di codifica per istruire la macchina sulle operazioni aritmetiche da compiere, "il lavoro aritmetico è stato sottratto al matematico, che è divenuto un programmatore, e questo compito è stato affidato all'UNIVAC". Ma ben presto, dati anche i costi del calcolatore e le esigenze di produzione, il lavoro di codifica "incombe allora come un nuovo fardello per il cervello umano". A liberare il cervello da questo peso interviene il catalogo di subroutine che Hopper aveva preparato per l'UNIVAC. Non sono più necessarie tavole di formule o la conoscenza delle procedure di codifica di queste, ma "deve solo essere capace di usare il catalogo per fornire al computer le informazioni sul proprio problema. L'UNIVAC, sulla base delle informazioni fornite dal matematico, sotto il controllo di una 'routine compilatrice di tipo A', usando le subroutine e il proprio codice di istruzioni, genera un programma". Così il "programmatore può tornare ad essere un matematico". Egli non ha più a che fare con quantità numeriche, in quanto le informazioni che riceve dal calcolatore "trattano di variabili e costanti

⁵⁸⁹ "What was required to go further was an interface that could be used for many different purposes. The 'universal' digital computer needed a 'universal' interface"

⁵⁹⁰ "Grace Hopper was a pioneer in all of these areas. She described her goal as freeing mathematicians to do mathematics. It was echoed years later in the HCI goal of freeing users to do their work".

in forma simbolica, insieme alle operazioni da eseguire su di esse". Nel sistema ridotto "ai suoi due componenti principali - il matematico e il computer", al primo resta il compito di direzione, liberato dal peso del calcolo e della traduzione di questo in istruzioni [Hopper, 1952, trad. mia]⁵⁹¹.

La filosofia alla base del lavoro di Hopper era quella per cui attraverso UNIVAC

Cercavamo di risolvere problemi e ottenere risposte. E credo che dovremmo in qualche modo tornare a quella fase. Cercavamo anche di fornire uno strumento utilizzabile dalle persone, non solo da programmatori o progettisti di linguaggi di programmazione. Ma da persone comuni, normali, che avevano problemi da risolvere. Alcuni erano ingegneri, altri uomini d'affari. E cercavamo di rispondere ai bisogni delle diverse comunità [Hopper, 1981, p.11 trad. mia]⁵⁹².

A questa idea di fondo di Grace Hopper si deve la caratteristica più nota di COBOL, anch'esso tuttora utilizzato: "la possibilità di utilizzare nomi di variabili lunghi che rendevano il linguaggio risultante simile all'inglese comune" [Ceruzzi, 2003, p. 92, trad. mia]⁵⁹³. Prendendo come esempio l'istruzione "IF", introdotta da FORTRAN, quello che in questo linguaggio poteva leggersi come:

"IF A > B", in COBOL era "IF EMPLOYEE-HOURS IS GREATER THAN MAXIMUM".

Questa caratteristica lo rendeva appetibile per i manager che leggevano i programmi ma non scrivevano in codice. COBOL rappresentava una tappa importante, sebbene "illusoria",

⁵⁹¹ ""The basic element of a production line is the input of raw material, controlled by human beings, possibly through instruments, supplied with machine tools; the operation produces an automobile, a rail, or a can of tomatoes."; ""A very old operation, is the solution of a mathematical problem. It fits the operational configuration: input of mathematical data, control by the mathematician, supplied with memory, formulas, tables, pencil, and paper; the brain carries on the arithmetic and produces results. It is the current aim to replace, as far as possible, the human brain by an electronic digital computer."; "The arithmetical chore has been removed from the mathematician, who has become a programmer, and this duty assigned to the UNIVAC"; "his duty now looms as an imposition on the human brain"; "The programmer may return to being a mathematician. He is supplied with a catalogue of subroutines. No longer does he need to have available formulas or tables of elementary functions. He does not even need to know the particular instruction code used by the computer. He needs only to be able to use the catalogue to supply information to the computer about his problem. The UNIVAC, on the basis of the information supplied by the mathematician, under the control of a "compiling routine of type A", using subroutines and its own instruction code, produces a program."; "Reducing again to the two-component system? the mathematician and the computer"".

⁵⁹² "We were trying to solve problems and get answers. And I think we should go back somewhat to that stage. We were also endeavoring to provide a means that people could use, not that program mers or programming language designers could use. But rather, plain, ordinary people, who had problems they wanted to solve. Some were engineers; some were business people. And we tried to meet the needs of the various communities".

⁵⁹³ "It was from Grace Hopper that COBOL acquired its most famous attribute, namely, the ability to use long character names that made the resulting language look like ordinary English".

nell'evoluzione del rapporto tra linguaggio naturale e linguaggio macchina [Cfr, Ceruzzi, 2003, p.92-93]⁵⁹⁴.

In riferimento ai linguaggi di programmazione e alla possibilità di implementarli in macchine virtuali, legati a questa dimensione simbolica - sebbene ancora nella prospettiva dell'intelligenza artificiale classica, secondo cui la macchina era potenzialmente capace di riprodurre tutte le attività cognitive umane - va menzionato il LISP (*LISt Processing*), linguaggio ideato da John McCarthy, lo stesso che nel 1956 coniò il termine "intelligenza artificiale" in occasione del celebre Summer Research Project di Dartmouth. Sin da prima di questa occasione McCarthy iniziò a riflettere sulla necessità di un linguaggio in grado di rappresentare e manipolare strutture simboliche complesse, come quelle richieste da un sistema in grado di simulare i processi cognitivi umani, di un linguaggio per l'intelligenza artificiale [Cfr. Stoyan, 1984; Gardner, 1985, pp. 177-178 trad. it.].

LISP fu implementato al MIT nel 1958 e pubblicato nel 1960 nell'articolo *Recursive functions of symbolic expressions and their computation by machine, Part* [McCarthy, 1960]. In LISP tanto i programmi quanto i dati erano rappresentati come liste, o liste di liste, secondo la notazione detta S-espressioni [*S-expressions*] che rispecchiavano, pienamente in linea con il paradigma GOF AI, la rappresentazione interna del codice e dei dati nella memoria del calcolatore. Tra le sue caratteristiche più distintive si trovano l'impiego delle espressioni lambda (riprese da Alonzo Church), per la definizione di funzioni; l'uso ricorsivo delle espressioni condizionali per costruire funzioni calcolabili; e la possibilità per il linguaggio di auto-riferirsi e manipolare sé stesso, grazie alla funzione *eval* e alla rappresentazione dei programmi come dati [Cfr. McCarthy, 1981, pp.173-174]. LISP divenne presto il linguaggio più utilizzato nella ricerca sull'intelligenza artificiale, in quanto progettato per rispecchiare le strutture mentali e i passi della risoluzione simbolica dei problemi⁵⁹⁵.

Il linguaggio di programmazione di McCarthy era diretta espressione delle sue idee "filosofiche": "Un programma informatico capace di agire in modo intelligente nel mondo deve disporre di una rappresentazione generale del mondo, in base alla quale interpreta i propri input. Progettare un simile programma implica prendere posizione su cosa sia la conoscenza e

⁵⁹⁴ Su COBOL, per un'interessantissima analisi tecnica, storica e socio-psicologica, si veda Schneiderman, 1985. Grace Hopper racconta in prima persona il percorso che l'ha portata a lavorare su UNIVAC e sviluppare i compiler e poi COBOL in Hopper, 1981, pp.7-24. Sullo stesso tema si veda anche Beyer, 2009, 276-310.

⁵⁹⁵ Il rapporto tra LISP e i dettami dell'Intelligenza Artificiale simbolica, con un approfondimento tecnico del linguaggio di programmazione di McCarthy, è analizzato in Haugeland, 1985, pp.139-149 trad. it.

su come essa venga acquisita” [McCarthy, Hayes, 1969 trad. mia]⁵⁹⁶. Deve cioè poter inferire i suoi obiettivi e le sue modalità d’azione da questa sua rappresentazione del mondo espressa in linguaggio formale. Quest’idea necessita una presa di posizione radicale: che “tutta la conoscenza sia (o possa essere) concepita in termini puramente logici” [Gardner, 1985, p.178 trad. it.]. Questo obiettivo, secondo McCarthy “richiede la formalizzazione dei concetti di causalità, abilità e conoscenza” [McCarthy, Hayes, 1969 trad. mia]⁵⁹⁷.

Accanto a queste innovazioni nella programmazione bisogna segnalarne un’altra, un “che sostituì gli operatori umani incaricati di gestire il flusso di lavoro in un impianto basato su schede perforate: il *sistema operativo*” [Ceruzzi, 2012, p.63 trad. mia enfasi mia]⁵⁹⁸.

Se si considerano gli elementi fondamentali dell’architettura del calcolatore di von Neumann - memoria, unità di controllo, sistemi di bus e di I/O - il sistema operativo, concetto non di facile definizione, può essere inteso come lo “strato di software, il cui compito è fornire ai programmi utente un modello del computer migliore, più semplice e più ordinato, e gestire tutte le risorse appena menzionate” [Tanenbaum, Bos, 2022, p. 1, trad. mia]⁵⁹⁹. Il suo scopo è duplice: “mediare tra i programmi applicativi, le utility e gli utenti, da un lato, e l’hardware del sistema informatico, dall’altra” [Stallings, 2008, p.6 trad. mia]⁶⁰⁰.

L’hardware dei computer, con cui un tempo ci si interfacciava direttamente attraverso il linguaggio macchina, era troppo complesso e difficile da gestire, in particolare per le operazioni di input/output. Per ridurre questa complessità, il sistema operativo introduce strati di astrazione, come i file, che permettono ai programmi di manipolare i dati senza dover gestire i dettagli tecnici dell’hardware.

Per agevolare il lavoro del programmatore viene fornito un insieme di programmi di sistema, tra cui le *utility* - piccoli strumenti software che svolgono funzioni comuni, come la

⁵⁹⁶ “A computer program capable of acting intelligently in the world must have a general representation of the world in terms of which its inputs are interpreted. Designing such a program requires commitments about what knowledge is and how it is obtained”.

⁵⁹⁷ “This requires formalizing concepts of causality, ability, and knowledge. Such formalisms are also considered in philosophical logic”. *Some Philosophical Problems from the Standpoint of Artificial Intelligence* scritto da McCarthy e dal computer scientist britannico Patrick J. Hayes (1944- vivente) rappresenta una pietra miliare della filosofia dell’Intelligenza Artificiale classica-simbolica e del perché l’AI “ha bisogno della filosofia” [“Why Artificial Intelligence Needs Philosophy”, McCarthy, Hayes, 1969], come recita il titolo del secondo capitolo di questo saggio seminale.

⁵⁹⁸ “As commercial computers took on work of greater complexity, another type of program emerged that replaced the human operators who managed the flow of work in a punched card installation: the operating system”.

⁵⁹⁹ “For this reason, computers are equipped with a layer of software called the operating system, whose job is to provide user programs with a better, simpler, cleaner, model of the computer and to handle managing all the resources just mentioned”.

⁶⁰⁰ “An operating system mediates among application programs, utilities, and users, on the one hand, and the computer system hardware on the other”.

modifica del codice, la gestione dei file o il controllo dei dispositivi di input/output. Queste vengono utilizzate sia durante lo sviluppo delle applicazioni, sia durante la loro esecuzione, per compiere operazioni specifiche richieste dal programma. Il sistema operativo costituisce la collezione più importante di programmi di sistema e può essere visto come un insieme coordinato di utility fondamentali. Dal lato dell'utente, esso genera e gestisce astrazioni che nascondono la complessità dell'hardware, offrendo interfacce semplici, coerenti ed efficienti. Dal lato dell'hardware, invece, il sistema operativo amministra le risorse fisiche: assegna in modo ordinato il processore ai vari processi, ripartisce la memoria principale tra i programmi in esecuzione e coordina l'accesso ai dispositivi di input/output, garantendo efficienza e continuità operativa [Tanenbaum, Bos, 2023, pp. 5-7; Stallings, 2008, p. 51].

Alla figura - spesso femminile - che riceveva fogli di carta con astruse istruzioni scritte in notazione logica, e che doveva tradurle manualmente in configurazioni di spinotti, prese e cavi, oppure preparare nastri e schede perforate, si sostituisce un nuovo intermediario "virtuale": il sistema operativo. A esso spetta il duplice compito di gestire la dimensione fisica della macchina e di mediare la relazione con l'utente finale della computazione.

Antesignano dei sistemi operativi di computer e smartphone - come Windows, macOS, Linux, Android e iOS - che gestiscono automaticamente le risorse della macchina e offrono all'utente interfacce semplici per avviare applicazioni, salvare dati e accedere a dispositivi, fu il JCL (*Job Control Language*) sviluppato da IBM nel 1960. Il JCL consisteva in un insieme di comandi scritti su schede perforate - il cosiddetto *batch programming*, in cui i programmi venivano eseguiti automaticamente a partire da mazzi di schede consegnati in blocco - e serviva a indicare al computer come interpretare ciò che seguiva: se il mazzo successivo conteneva un programma scritto in Fortran o in un altro linguaggio, se si trattava di dati da elaborare, oppure se dovesse avviarsi il lavoro di un altro utente [Cfr. Ceruzzi, 2012, p. 63].

Ma per gestire questi flussi serviva un software di sistema adeguato: se progettato male, poteva introdurre un tale sovraccarico da compromettere l'efficienza del computer. E questo fu proprio il caso di JLC, il giudizio dal suo principale responsabile, il *computer scientist* e ingegnere del software, Fred Brooks (1931-2022) è lapidario: "Sono convinto che sia il peggior linguaggio di programmazione per computer mai concepito da chiunque, ovunque. È stato sviluppato sotto la mia supervisione; la colpa è abbastanza grande da poter essere condivisa a tutti i livelli della supervisione" [Brooks, 2010, p.169 trad. mia]⁶⁰¹

⁶⁰¹ I am convinced, the worst computer programming language ever devised by anybody, anywhere. It

Brooks è stato anche l'autore di *The Mythical Man-Month* (1975), definito la “bibbia dell'ingegneria del software”. Il titolo si basa sull'idea del *man-month* (uomo-mese), un'unità di misura che considera tempo e persone come intercambiabili. Brooks dimostrò invece come, nello sviluppo software, questa equivalenza costituisca un mito pericoloso: l'aggiunta di programmatori non riduce i tempi, ma amplifica errori e difetti del risultato finale. È questo il cuore della celebre Legge di Brooks: “Aggiungere manodopera a un progetto software in ritardo lo fa ritardare ancora di più” [Brooks, 1975, p. 25, trad. mia]⁶⁰². Queste considerazioni costituiscono un esempio della crescente importanza della figura - e della definizione dei compiti - del *designer*, la cui centralità si era già iniziata a delineare con l'esperienza di Monte Carlo. Nel 1986, Brooks sosterrà che non esiste nessuna formula magica o “proiettile d'argento” [*silver bullet*] capace di risolvere in un colpo solo i problemi di complessità del software, dovuti alla stessa natura di quest'ultimo [Brooks, 1986]. L

Il tipico progetto software ha qualcosa di questa natura (almeno dal punto di vista del manager non tecnico) [...] è capace di trasformarsi in un mostro fatto di scadenze mancate, budget sfiorati e prodotti difettosi. Così si levano grida disperate alla ricerca di un *silver bullet*, una “pallottola d'argento” capace di far crollare i costi del software con la stessa rapidità con cui scendono quelli dell'hardware. Ma, se guardiamo all'orizzonte del prossimo decennio, non vediamo alcuna pallottola d'argento. Non esiste alcuno sviluppo, né tecnologico né gestionale, che da solo prometta un miglioramento anche solo di un ordine di grandezza in termini di produttività, affidabilità o semplicità [Brooks, 1986 trad. mia]⁶⁰³.

was developed under my supervision; there is blame enough to go around among all the supervisory levels”. Per il funzionamento, e il malfunzionamento, del JLC si veda Brooks, 2010, pp.169-173.

⁶⁰² Oversimplifying outrageously, we state Brooks's Law: Adding manpower to a late software project makes it later. Per Brooks, far funzionare davvero un programma non significa solo farlo compilare o eseguire, ma progettarlo in modo chiaro, comprensibile e mantenibile. Tra le sue proposte figurano: affidare la progettazione a pochi architetti, un “team chirurgico” [surgical team, Brooks, 1975, pp.29-37], per garantire coerenza concettuale (Brooks, 1975, pp.41-46).; scrivere documentazione chiara [Brooks, 1975, pp.73-82]; costruire prima un “prototipo di sistema” da buttare via perché “si finirà comunque per scriverne uno che non funziona, [Hence plan to throw one away; you will, anyhow, Brooks, 1975, p.116]; e incentivare la comunicazione continua tra membri del team, spesso più importante delle capacità tecniche individuali. Brooks sottolinea anche che molti fallimenti derivano da aspettative irrealistiche, da una cattiva gestione del tempo e da una sottovalutazione della complessità [Brooks, 1975, pp.18-19].

⁶⁰³ “The familiar software project has something of this character (at least as seen by the non-technical manager), usually innocent and straightforward, but capable of becoming a monster of missed schedules, blown budgets, and flawed products. So we hear desperate cries for a silver bullet, something to make software costs drop as rapidly as computer hardware costs do. But, as we look to the horizon of a decade hence, we see no silver bullet. There is no single development, in either technology or management technique, which by itself promises even one order of magnitude improvement in productivity, in reliability, in simplicity”.

JLC fu implementato su un calcolatore lanciato da IBM nell'aprile del 1964, sul quale l'azienda "scommise la compagnia": il System/360, così chiamato perché progettato per coprire l'intero spettro delle esigenze computazionali - dalla matematica complessa all'aritmetica più semplice, sia scientifica che commerciale - a 360 gradi, da cui appunto il nome del sistema [Cfr. Ceruzzi, 2003, p. 144].

Il System/360 non era un singolo computer, ma una famiglia di macchine, pensata per sostituire i prodotti IBM preesistenti in tutte le fasce di mercato. "Poiché ogni modello condivideva lo stesso set di istruzioni (con alcune eccezioni), il software scritto per un modello di fascia bassa poteva essere trasferito su un modello di fascia più alta man mano che crescevano le esigenze del cliente, preservando così l'investimento fatto nei programmi sviluppati" [Ceruzzi, 2012, p. 92, trad. mia]⁶⁰⁴.

Adottando l'approccio del *microprogramming* (pionieristicamente introdotto da M. Wilkes con l'EDSAC), tecnica che consiste nel costruire la sezione di controllo come una sorta di piccolo computer capace di eseguire micro-istruzioni codificate in memoria, IBM rese possibile emulare - termine che, nelle intenzioni del management dell'azienda, indicava la riproduzione fedele del comportamento di una macchina diversa, e non una semplice imitazione o simulazione - i programmi delle macchine precedenti, come l'IBM 1401 e l'IBM 7070, garantendo così compatibilità e continuità d'uso attraverso tutta la famiglia System/360 [Cfr. Haigh, Ceruzzi, 2021, p. 68].

L'architettura, di tipo *general purpose* e a programma memorizzato, prevedeva parole di 32 bit, 16 registri generali e un indirizzamento a 24 bit, che consentiva l'accesso diretto a una memoria principale - enorme per l'epoca - di 16 megabyte, realizzata con nuclei magnetici, con tempi d'accesso variabili tra 750 nanosecondi e 8 microsecondi. Il sistema supportava sia dati binari che decimali ed era affiancato da dispositivi come nastri magnetici, dischi rigidi e schede perforate [Cfr. Amdahl et al., 1964]. IBM standardizzò inoltre l'uso di byte da 8 bit, capaci di rappresentare 256 simboli, adottando il codice EBCDIC. Questa scelta, però, entrava in contrasto con lo standard ASCII, che nel 1963 fissava la codifica dei caratteri su 7 bit per ragioni meccaniche legate all'uso del nastro perforato.

⁶⁰⁴ . The System/360 was not just a single computer but a family of machines, from an inexpensive model intended to replace the popular IBM 1401 to high-end computers that were optimized for numerical calculations. Because each model had the same instruction set (with exceptions), software written for a small model could be ported to a higher-end model as a customer's needs grew, thus preserving customers' investment in the programs they had developed".

IBM investì ingenti risorse nello sviluppo di un nuovo sistema operativo e di un nuovo linguaggio di programmazione (PL/I), che però si rivelarono meno efficaci del previsto. Per fortuna, le estensioni di software già consolidati come FORTRAN e COBOL, insieme a sistemi operativi sviluppati al di fuori della linea ufficiale IBM, riuscirono a sostenere il progetto, salvando un investimento considerato il più grande mai fatto da un soggetto privato al di fuori dell'ambito governativo. La scommessa, alla fine, pagò. Entro la fine degli anni Sessanta IBM aveva vinto. Tra il 1965 e il 1970, grazie al System/360, i ricavi lordi dell'azienda più che raddoppiarono, così come gli utili netti, che superarono il miliardo di dollari nel 1971.

Entro il 1970, IBM contava 35.000 computer installati e dominava il mercato statunitense a tal punto che si cominciò a parlare, nel confronto con la concorrenza, di IBM da una parte, e di tutti gli altri insieme - i "sette nani" [*the seven dwarves*] - dall'altra. Una supremazia così netta da portare all'apertura di un procedimento antitrust federale [Cfr. Ceruzzi, 2003, p. 144; Ceruzzi, 2012, p. 94].

3. California dreamin': un viaggio elettronico verso la Silicon Valley

Il successo commerciale dell'IBM - di cui von Neumann era consulente - insieme al dibattito da lui promosso sull'impiego delle macchine veloci per affrontare nuovi campi scientifici (dall'idrodinamica alla meccanica quantistica fino alle procedure di Monte Carlo) [Cfr. von Neumann, 1951], veicolava - ed era sostenuto, come abbiamo visto - l'idea, tutt'altro che neutrale dal punto di vista sociotecnico, della metafora cibernetica.

L'analogia funzionale tra cervello e macchina, e lo sviluppo conseguente del calcolatore, devono ora essere letti alla luce dei progressi dell'elettronica a stato solido, che ne hanno permesso l'effettiva realizzazione. Questo passaggio segna nuovi mutamenti di paradigma, con ricadute sia sulla materializzazione della computazione sia sul rapporto tra macchina ed essere umano, così come sulle modalità di implementazione delle funzioni logiche. Da qui prende forma la traiettoria tecnologica e sociotecnica che dalle macchine finora considerate, conduce alla nascita della computazione personalizzata: quella dinamica scientifica e culturale incontrata all'inizio del nostro lavoro, che ora trova concreta realizzazione nell'analisi della dimensione hardware.

"I tubi a vuoto, al posto di dispositivi meccanici più lenti, furono fondamentali per l'invenzione del computer digitale a programma memorizzato. Lo stesso fenomeno si ripeté di

nuovo, intorno al 1960, con l'introduzione del transistor a stato solido come sostituto dei tubi" [Ceruzzi, 2012, p.64 trad. mia]⁶⁰⁵.

I *vacuum tubes*, come più volte sottolineato, pur essendo di gran lunga più veloci dei dispositivi elettromeccanici, erano componenti caratterizzate da una grande inaffidabilità. "Per tutta la prima metà del Novecento, fisici e ingegneri elettrici cercarono modi per sostituire il tubo con un dispositivo in grado di commutare o amplificare correnti in un pezzo di materiale solido, senza vuoto o filamento caldo" [Ceruzzi, 2012, p.66 trad. mia]⁶⁰⁶.

Il problema della sostituzione delle valvole con dispositivi più avanzati, pur trovando un mercato potenzialmente pronto ad accoglierli, era ostacolato dalla dimensione teorica: la comprensione ancora incompleta della fisica quantistica alla base della fisica dello stato solido rallentava infatti la possibilità di tradurre tali innovazioni in realizzazioni concrete [Brinkman et. al., 1997, p.1857]⁶⁰⁷ Trovare un sostituto delle valvole si rivelava un compito frustrante nello sviluppo ulteriore della computazione elettronica.

Il passo cruciale si compì appena dopo la fine della guerra. Pur se incredibilmente, col senno di poi, sottovalutato - il *New York Times* del 1° luglio 1948 ne diede notizia in una nota modesta nella rubrica *The News of Radio* [New York Times, 1948] - la vigilia del Natale del 1947 [Shockley, 1984] i Bell Labs avevano prodotto il "primo dispositivo a stato solido funzionante come interruttore o amplificatore", battezzato "transistor" [Ceruzzi, 2012, p. 67 trad. mia]⁶⁰⁸. I tre inventori furono William Shockley (1910–1989), Walter Brattain (1902–1987) e John Bardeen (1908–1991) che per questa invenzione ricevettero il premio Nobel nel 1956⁶⁰⁹.

Il transistor funziona grazie al comportamento particolare degli elettroni in elementi noti come semiconduttori. In questi materiali, come germanio e silicio, la corrente può scorrere non solo attraverso il movimento degli elettroni carichi negativamente, ma anche tramite le "lacune" [*holes*], ossia le assenze di elettroni che si comportano come cariche positive mobili.

⁶⁰⁵ "We saw how vacuum tubes in place of slower mechanical devices were crucial to the invention of the stored program digital computer. The phenomenon was repeated again, around 1960, by the introduction of the solid-state transistor as a replacement for tubes".

⁶⁰⁶ "Through the early twentieth century, physicists and electrical engineers sought ways of replacing the tube with a device that could switch or amplify currents in a piece of solid material, with no vacuum or hot filament required".

⁶⁰⁷ "Without quantum mechanics, we would never have developed any comprehensive understanding of solids. The understanding of the differences between metals, insulators, and semiconductors was quickly developed. The concept of electronic band structure, due to quantum mechanics, was the key to that insight".

⁶⁰⁸ "After the war, a concentrated effort at Bell Laboratories led to the invention of the first successful solid-state switching or amplifying device, which Bell Labs christened the "transistor."".

⁶⁰⁹ La storia del travagliato cammino che ha portato all'invenzione del transistor è narrata in prima persona da William Shockley in Shockley, 1984. Intorno alla storia di questa fondamentale componente tecnologica si veda anche Riordan et al. 1999.

Questa duplice possibilità di conduzione rese i semiconduttori molto più versatili dei metalli. Il primo transistor, costruito nel 1947 in germanio, era formato da due sottili contatti puntati su un cristallo: esternamente ricordava i rivelatori “a baffo di gatto” delle vecchie radio, ma al suo interno si basava su una nuova comprensione teorica dei processi quantistici. Fu questo ponte tra teoria quantistica ed implementazione a gettare tra il 1947 e il 1960 le basi della rivoluzione dell’elettronica dello stato solido.

Il transistor a contatto puntiforme del 1947 fu presto sostituito da dispositivi ottenuti unendo semiconduttori di tipo diverso. Si svilupparono tecniche per purificare germanio e soprattutto silicio, introducendo impurità controllate, un processo noto come “drogaggio” o *doping*) che rendevano il materiale di tipo P (conduzione tramite lacune, le cariche positive) o di tipo N (conduzione tramite elettroni, ossia le cariche negative). Grazie alla fotolitografia e all’uso di strati isolanti impiegando il silicio [*silicon*] divenne infine possibile la miniaturizzazione elettronica. [Bardeen, Brattain, 1949; Cfr. Ceruzzi, 2012, p.68].

Il fatto che il transistor fu sviluppato dai Bell Labs e annunciato pubblicamente è significativo: non essendo vincolati dalla segretezza militare, poterono offrire l’invenzione al mercato. Tuttavia, in quanto parte di un monopolio regolamentato, i laboratori si mossero con cautela ed evitarono di entrare nel settore dei computer commerciali, finché nel 1956 un decreto consensuale con il Dipartimento di Giustizia degli USA li obbligò a limitarsi alle comunicazioni come *common carrier*, cioè come fornitore pubblico di servizi telefonici sotto stretto controllo statale [Cfr. Ceruzzi, 2003, p.64-65].

Queste vicissitudini contribuirono a far sì che l’impiego del transistor nei calcolatori si affermasse soltanto alla fine degli anni Cinquanta, con due modelli IBM che consolidarono la posizione dell’azienda sul mercato. L’IBM 1401, pensato per integrarsi con le installazioni a schede perforate, rese più semplice per molti clienti commerciali il passaggio all’elettronica e divenne un enorme successo di vendita [Bashe et al., 1986 pp. 472-474; Haigh, Ceruzzi, 2021, pp.58-59]. L’IBM 7090, che è ben visibile nel film di Stanley Kubrick del 1964 *Il dottor Stranamore*, invece, nacque da un grande sforzo di riprogettazione del modello 709 a valvole: su richiesta dell’aeronautica americana, che voleva un sostituto del SAGE in chiave antisovietica, l’intera architettura fu riconfigurata con grande sforzo in una a transistor al posto di quella a tubi a vuoto, dando vita a un calcolatore più veloce e affidabile che trovò ampia applicazione in ambito scientifico, militare e aerospaziale [Pugh et al., 1991, pp.36-37; Ceruzzi, 2012, p. 69-70; Haigh, Ceruzzi, 2021, p.57].

L'introduzione e la diffusione del transistor consentono di riflettere sul tema del "determinismo tecnologico", secondo cui lo sviluppo di una tecnologia costituirebbe il motore inesorabile dei cambiamenti sociali. Emblematica in tal senso è la già citata *Legge di Moore* [Moore, 1965], secondo cui la complessità di un microcircuito - misurata ad esempio dal numero di transistor per chip - raddoppia ogni 18 mesi. Tuttavia, "per oltre un decennio prima del 1960 la legge di Moore aveva una pendenza pari a zero: il numero di transistor su un chip restava pari a uno. Portare il dispositivo a uno stato in cui se ne potessero collocare più di uno su un singolo pezzo di materiale, facendoli funzionare in modo affidabile e coerente, richiese uno sforzo concentrato" [Ceruzzi, 2012, pp. 66-67, trad. mia]⁶¹⁰.

I calcolatori incontrati finora sono quelli definiti *mainframe*: grandi, o in alcuni casi, come abbiamo visto, giganteschi, computer, originariamente ospitati in imponenti telai metallici da cui deriva il loro nome, progettati per gestire enormi quantità di dati e numerosi utenti simultanei. Nati negli anni Cinquanta con migliaia di valvole, poi sostituite dai transistor, occupavano intere stanze attrezzate con sistemi dedicati di raffreddamento e cablaggio [Cfr. Ceruzzi, 2012, p.54; Meek, in Reilly, 2004 pp. 481-482].

La diffusione dell'elettronica a stato solido ha provocato un "cambio di paradigma" che ha portato a una mutazione della struttura stessa del calcolatore e della sua relazione con l'utente umano. "L'IBM 7090 era un computer a transistor, come la radio era un "telegrafo senza fili" o l'automobile una "carrozza senza cavalli": conservava il modello mentale dei computer esistenti, solo con una nuova e migliore tecnologia di base. Ma cosa succederebbe se si partisse da un foglio bianco e si progettasse una macchina che sfruttasse il transistor per quello che è, e non per ciò che va a sostituire?" [Ceruzzi, 2012, p.70 trad. mia]⁶¹¹.

A questa domanda risposero Kenneth H. Olsen (1926-2011) e Harlan E. Anderson (1929-2019), entrambi provenienti dai *Lincoln Laboratories*, dove avevano lavorato al TX-0, un prototipo basato sui transistor ad alta velocità prodotti dalla *Philco* [Ceruzzi, 2003, p.65] e destinato a testarne la fattibilità su larga scala. Con soli 70.000 dollari di capitale di rischio

⁶¹⁰ "However, we must remember that for over a decade prior to 1960, Moore's law had a slope of zero: the number of transistors on a chip remained at one. Bringing the device to a state where one could place more than one of them on a piece of material, and get them to operate reliably and consistently, took a concentrated effort".

⁶¹¹ "The IBM 7090 was a transistorized computer like the radio was a "wireless" telegraph or the automobile was a "horse less" carriage: it retained the mental model of existing computers, only with a new and better base technology. What if one started with a clean sheet of paper and designed a machine that took advantage of the transistor for what it was, not for what it was replacing?"

fondarono la *Digital Equipment Corporation* (DEC)⁶¹², in un piccolo ex-opificio tessile a Maynard nel Massachusetts, quasi plasticamente l'antitesi di un colosso come IBM.

Per distinguersi dai grandi mainframe, chiamarono i loro strumenti *Programmed Data Processor* (PDP), evitando volutamente il termine "computer" che evocava una sfida diretta a IBM. Non si trattava più di enormi macchine centralizzate, ma di sistemi più piccoli ed economici, capaci di stare accanto a un banco da laboratorio e di essere utilizzati in modo interattivo [Ceruzzi, 2012, p.70-71, Haigh, Ceruzzi, 2021, pp.91-92].

Il primo prodotto della DEC fu il PDP-1 [fig.24], presentato nel 1959. Riprendeva molte innovazioni del sistema SAGE, tra cui il display a tubo catodico e la penna ottica che permettevano una vera interazione in tempo reale con i programmi. A differenza dei mainframe "adattati" dell'IBM, il PDP-1 era progettato fin dall'inizio attorno alle nuove possibilità offerte dallo stato solido: poteva eseguire 100.000 addizioni al secondo e disponeva di una memoria a nuclei magnetici di 4.000 parole, espandibile fino a 64.000. Venduto a 120.000 dollari, risultava molto più economico rispetto ai grandi sistemi IBM, ed ebbe clienti di rilievo come la Bolt, Beranek e Newman (BBN), il MIT o il Lawrence Livermore Laboratory, che lo impiegarono in applicazioni interattive e sperimentali, dal text editing ai primi videogiochi [Haigh, Ceruzzi, 2021, pp.91-92]⁶¹³.

Il PDP-1 "portò il calcolo fuori dalla sala computer e nelle mani dell'utente" [Ceruzzi, 2012, p.71 trad. mia]⁶¹⁴. I computer della DEC "aprono un'enorme strada verso il futuro: dal mondo dei mainframe a elaborazione a lotti di IBM, verso il mondo che conosciamo oggi, fatto di dispositivi piccoli, economici e utilizzabili in modo interattivo" [Ceruzzi, 2012, p. 71 trad. mia]⁶¹⁵.

Questa "rivoluzione" continuò nel 1965 con il lancio del PDP-8 [fig.25]. Costava, un prezzo senza precedenti per l'epoca, solo 18.000 dollari nella versione base ed era abbastanza compatto da poter essere installato accanto a normali *rack* di laboratorio, senza bisogno di alimentazioni dedicate o impianti di climatizzazione. La macchina occupava circa 0,23 metri cubi di volume e pesava circa 113 chilogrammi. Il cuore del calcolatore era costituito da

⁶¹² Sulla figura di Olsen e sulla storia della DEC si rimanda a Rifkin, 1990 e a Earls, 2004.

⁶¹³ I report sui progressi dell'implementazione del PDP-1 della DEC si possono trovare in Bell, G. (1998, 30 aprile). *PDP-1 Story*. Digital Equipment Corporation. URL: gordonbell.azurewebsites.net/digital/timeline/pdp-1story.htm

⁶¹⁴ "Brought computing out of the computer room and into the hands of the user"

⁶¹⁵ "Olsen and Anderson did not see the path that computers would ultimately take, and Olsen in particular would be criticized for his supposed lack of foresight. But they opened a huge path to the future: from the batch-oriented mainframes of IBM to the world we know today: a world of small, inexpensive devices that could be used interactively".

transistor di ultima generazione, montati su moduli logici standard e collegati con la tecnica automatizzata del *wire wrap*, che sostituiva le saldature manuali. L'architettura a 12 bit⁶¹⁶, molto più semplice rispetto a quella dei mainframe, consentiva di ridurre drasticamente costi e complessità, aprendo la strada a un uso più diffuso del calcolatore in ambiti scientifici, industriali e di controllo [Haigh, Ceruzzi, 2021, p.93; Ceruzzi, 2012, p.72]

Fig. 24 e 25: il PDP-1 e il PDP-8



Con il PDP-1 prima e soprattutto con il PDP-8, Olsen e Anderson mostrarono che il transistor non era solo un rimpiazzo delle valvole, ma l'occasione per inventare un nuovo modello di calcolatore, flessibile, modulare e vicino all'utente finale. Ma le innovazioni dei modelli PDP non erano solo architetture. Rispecchiando la loro politica di accesso libero all'informazione da parte dell'utente [Haigh, Ceruzzi, 2021, p.93]⁶¹⁷, avevano avviato una scintilla che avrebbe cambiato il corso della computazione portandola, a livello hardware, ad un rapporto *personalizzato* con l'essere umano.

⁶¹⁶ Le specifiche architetture si trovano in Haigh, Ceruzzi, 2021, p.93.

⁶¹⁷ "Yet the simplicity of the PDP- 8's architecture, coupled with DEC's policy of making information freely available, made it an easy computer to understand".

Priva delle risorse economiche di titani come l'IBM, la DEC "incoraggiava i clienti stessi a sviluppare l'hardware e il software specializzati. Condivideva i dettagli del design e delle caratteristiche operative del PDP-8 e collaborava con gli utenti per integrare le macchine come controllori per l'automazione industriale, la commutazione telefonica, la strumentazione biomedica e altri usi" [Ceruzzi, 2012, pp.72-73 trad. mia]⁶¹⁸

La natura "materiale" *general-purpose* dei calcolatori PDP, resa possibile dall'innovazione del transistor e da un design che ne sfruttava appieno le potenzialità, li rendeva adattabili a qualunque scopo computazionale. In questo senso essi rappresentavano una realizzazione pratica del progetto avviato da Turing con *On Computable Numbers*: quello di una macchina universale [Ceruzzi, 2012, p. 74]. John Leng, dirigente britannico di alto livello della DEC, in un resoconto di vendita scriveva: «Ecco le ultime attività dei minicomputer nella terra delle minigonne mentre giro a bordo della mia Mini Minor». L'espressione ebbe fortuna e il nome prese piede [Rifkin et al., 1983, p. 72]. Accanto a queste innovazioni sociali, ne era nata un'altra "mini", il *minicomputer*: "L'era dell'informatica personale e interattiva non era ancora arrivata, ma si stava avvicinando" [Ceruzzi, 2012, p.74 trad. mia]⁶¹⁹.

Il PDP-1, di cui un prototipo era stato venduto alla Bolt Beranek and Newman, aveva attratto l'attenzione di un suo dirigente di alto livello, lo psicologo J. C. R. Licklider (1915-1990)⁶²⁰, già partecipe delle Macy Conferences sulla cibernetica con un intervento nel 1950 dal

⁶¹⁸ "At the highest level, DEC had initiated a social innovation as significant as the technology. DEC was a small company and did not have the means to develop specific applications for its customers, as IBM's sales force was famous for doing. Instead it encouraged the customers themselves to develop the specialized systems hardware and software. It shared the details of the PDP-8's design and operating characteristics and worked with customers to embed the machines into controllers for factory automation, telephone switching, biomedical instrumentation, and other uses".

⁶¹⁹ "The era of personal, interactive computing had not yet arrived, but it was coming".

⁶²⁰ Per la vicenda biografica e il ruolo chiave svolto da Licklider nello sviluppo dell'informatica – dal progetto SAGE alla nascita di ARPANET, l'antesignana della moderna Internet si rimanda a Waldrop, 2001. Un contributo significativo in lingua italiana intorno al pensiero dello psicologo statunitense intorno alla natura del rapporto tra uomo e calcolatore e sull'influenza che avrà nello sviluppo del Web si trova in Castellucci, 2000. Altrettanto importante è l'analisi contenuta in Numerico et al., 2010, pp. 37-5), che ricostruisce le idee di "Lick" (come era soprannominato lo scienziato americano) dalle prime intuizioni sulla simbiosi uomo-macchina sino ai sogni delle *biblioteche del futuro* [*libraires of the future*] [Licklider, 1965] e della rete intergalattica [*Intergalactic Computer Network*]. [Licklider, 1963]. Per Licklider, la biblioteca non sarebbe più stata soltanto un luogo fisico destinato alla conservazione dei libri, ma un "centro pensante" capace di offrire a molti utenti, anche a distanza, un accesso immediato all'intero patrimonio di conoscenze. L'immenso potere delle macchine di calcolo, che avrebbe reso disponibili le risorse a più postazioni contemporaneamente, avrebbe consentito di digitalizzare, organizzare e rendere consultabili i materiali, trasformando la biblioteca in un'infrastruttura informatizzata per la produzione e la diffusione del sapere. Strettamente connesso con questa visione, nell'aprile del 1963, sarebbe emerso il celebre progetto della "rete intergalattica": un sistema di calcolatori interconnessi e cooperanti, in cui i ricercatori avrebbero potuto utilizzare indifferentemente dati e programmi presenti su macchine diverse, superando i limiti delle singole installazioni. In quella lettera circolare, Licklider avrebbe invitato i colleghi a unire gli sforzi per costruire una rete che non fosse solo uno strumento tecnico, ma una nuova infrastruttura di comunicazione e collaborazione tra esseri umani mediata dai computer.

titolo *The Manner in Which and Extent to Which Speech Can Be Distorted and Remain Intelligible* [Licklider, 1950, pp.203-247].

Nel novembre 1962 egli prese parte a una conferenza sulle “scienze dei sistemi informativi”, organizzata dall’Aeronautica statunitense e dalla MITRE Corporation, *un think tank* finanziato dal governo per supportare la difesa. L’incontro si svolse all’indomani della crisi dei missili di Cuba, quando la scarsità di comunicazioni non solo tra Stati Uniti e URSS ma anche tra Casa Bianca, Pentagono e comandi navali aveva reso ancora più drammatica l’escalation. Da poco nominato direttore dell’*Information Processing Techniques Office* (IPTO) presso l’*Advanced Research Projects Agency* (ARPA), l’agenzia fondata nel 1958 (insieme alla NASA) in risposta al lancio dello Sputnik [Grundin, 2017, p.20; Ceruzzi, 2012, p.75; Numerico et al., 2010, p.37], Licklider rimase deluso dalla conferenza. “Aveva ascoltato molte relazioni su come i computer digitali potessero migliorare le operazioni militari, ma nessuno dei relatori, a suo avviso, aveva colto il vero potenziale dei computer nelle attività militari-o civili” [Ceruzzi, 2012, p.75 trad. mia]⁶²¹.

Tuttavia nel lungo viaggio ferroviario, dalle montagne Allegheny, in Virginia verso Washington, Licklider impiegò il tempo per riflettere con i colleghi sul vero potenziale dei computer, che egli vedeva andare ben oltre le applicazioni militari discusse in conferenza. Lo psicologo statunitense aveva dalla sua due caratteristiche cruciali: la disponibilità economica dovuta al suo ruolo governativo, “come direttore dell’IPTO, aveva accesso a grandi fondi del Dipartimento della Difesa e ampia libertà nel destinarli a progetti a sua discrezione” [Ceruzzi, 2012, p.76 trad. mia]⁶²² e, soprattutto, una visione ben chiara del ruolo e delle potenzialità del calcolatore su cui è opportuno soffermarsi. “A differenza di molti suoi pari, vedeva il computer digitale elettronico come un dispositivo rivoluzionario, non tanto per le sue capacità matematiche, quanto perché poteva essere utilizzato in *simbiosi*-il suo termine preferito-con gli esseri umani” [Ceruzzi, 2012, p.76 trad. mia]⁶²³.

Le idee di Licklider sulla simbiosi – che abbiamo già incontrato “potenzialmente” nelle grandi macchine, come ENIAC o Colossus - sono state espresse nel suo articolo del 1960 *Man-*

⁶²¹ “The conference in fact had been a disappointment. He had heard a number of papers on ways digital computers could improve military operations, but none of the presenters, in his view, recognized the true potential of computers in military-or civilian-activities”.

⁶²² “Licklider had two things that made his presence among that group critical. The first was money: as director of IPTO, he had access to large sums of Defense Department funds and a free rein to spend them on projects as he saw fit”.

⁶²³ “The second was a vision: unlike many of his peers, he saw the electronic digital computer as a revolutionary device not so much because of its mathematical abilities but because it could be used to work in symbiosis-his favorite term-with human beings”.

Computer Symbiosis [Licklider, 1960]. In questo paper il direttore dell'IPTO espone la sua concezione del ruolo del calcolatore in relazione all'essere umano, Collocandosi all'interno del paradigma cibernetico [Numerico et al., 2010, pp.38-39] condivideva con Wiener l'idea che lo sviluppo dei computer e dell'automazione non dovesse portare semplicemente ad "un uomo meccanicamente esteso" [Licklider, 1960]⁶²⁴.

Nei sistemi uomo-macchina del passato, secondo Licklider, il calcolatore e le parti meccaniche del sistema "erano meri prolungamenti, prima del braccio umano, poi dell'occhio umano". Questa estensione automatizzata ha portato come conseguenza che gli nei grandi sistemi di calcolo, "gli uomini che rimangono sono lì più per aiutare che per essere aiutati. In alcuni casi, in particolare nei grandi sistemi di informazione e controllo incentrati sui calcolatori, *gli operatori umani sono responsabili principalmente delle funzioni che si è rivelato non fattibile automatizzare*". All'essere umano - e qui non può sfuggire l'analogia dei risultati e delle conseguenze del lavoro di Turing sulla definizione rigorosa di algoritmo - rimaneva il compito di fornire "l'iniziativa, la direzione, l'integrazione e il criterio" [Licklider, 1960 trad. mia enfasi mia]⁶²⁵.

L'obiettivo di Licklider è differente. Impiegando il concetto, definito su basi strettamente biologiche, di *simbiosi*⁶²⁶ intesa come "vivere insieme in associazione intima, o addirittura in stretta unione, di due organismi dissimili", quello che lo psicologo statunitense intendeva realizzare era "un accoppiamento molto stretto tra i membri umani ed elettronici della partnership". Gli obiettivi di questa relazione erano: "1) consentire ai calcolatori di facilitare il pensiero formulativo così come oggi facilitano la soluzione di problemi già formulati, e 2)

⁶²⁴ "As a concept, man-computer symbiosis is different in an important way from [...] "mechanically extended man".

⁶²⁵ "In the man-machine systems of the past, the human operator supplied the initiative, the direction, the integration, and the criterion. The mechanical parts of the systems were mere extensions, first of the human arm, then of the human eye. [...] In one sense of course, any man-made system is intended to help man, to help a man or men outside the system. If we focus upon the human operator within the system, however, we see that, in some areas of technology, a fantastic change has taken place during the last few years. "Mechanical extension" has given way to replacement of men, to automation, and the men who remain are there more to help than to be helped. In some instances, particularly in large computer-centered information and control systems, the human operators are responsible mainly for functions that it proved infeasible to automate. Such systems ("humanly extended machines," North might call them) are not symbiotic systems."

⁶²⁶ Scrive Licklider nel definire la simbiosi: Il fico è impollinato solo dall'insetto *Blastophaga grossorum*. La larva dell'insetto vive nell'ovario del fico, e lì si nutre. L'albero e l'insetto sono dunque fortemente interdipendenti: l'albero non può riprodursi senza l'insetto; l'insetto non può nutrirsi senza l'albero; insieme, costituiscono non solo una partnership vitale, ma produttiva e fiorente. Questo "vivere insieme in associazione intima, o addirittura in stretta unione, di due organismi dissimili" viene chiamato simbiosi" ["The fig tree is pollinated only by the insect *Blastophaga grossorum*. The larva of the insect lives in the ovary of the fig tree, and there it gets its food. The tree and the insect are thus heavily interdependent: the tree cannot reproduce without the insect; the insect cannot eat without the tree; together, they constitute not only a viable but a productive and thriving partnership. This cooperative "living together in intimate association, or even close union, of two dissimilar organisms" is called symbiosis" Licklider, 1969 trad. mia.

permettere a uomini e calcolatori di cooperare nel prendere decisioni e nel controllare situazioni complesse senza una dipendenza inflessibile da programmi predeterminati". Ai due poli della relazione simbiotica spettano per Licklider compiti diversi, gli esseri – gli organismi - umani "stabiliranno gli obiettivi, formuleranno le ipotesi, determineranno i criteri ed effettueranno le valutazioni", mentre le macchine - gli organismi artificiali – "eseguiranno il lavoro routinario necessario a preparare la strada per intuizioni e decisioni nel pensiero tecnico e scientifico" [Licklider, 1960 trad. mia]⁶²⁷.

I sistemi classici di interazione uomo-macchina che Licklider ha per obiettivo polemico, non sono "simbiotici", sono, tutt'al più, "semi-automatici", "partiti con l'obiettivo di essere completamente automatici ma non hanno raggiunto tale traguardo" [Licklider, 1960 trad. mia]⁶²⁸. In essi vi "era un solo tipo di organismo - l'uomo - e il resto era lì soltanto per assisterlo" [Licklider, 1960 trad. mia]⁶²⁹, con la speranza di sostituirlo via via. Al contrario, il paradigma proposto da Licklider, che lo inserisce a pieno titolo nel programma di ricerca cibernetico - basti pensare ai sistemi integrati alla base del concetto di predittore antiaereo - era, come osserva T. Numerico, quello per cui "la macchina era invece partecipe e complice delle sue attività, surrogando l'operatore umano solo nelle componenti automatiche del compito e supportandolo nella presa di decisioni, senza mai sostituirlo" [Numerico et al., 2010, p. 40].

Licklider ci porta nel cuore della realizzazione di questa sua visione. I calcolatori – e le lunghe lotte delle operatrici di ENIAC o delle WREN per tradurre in configurazioni di cavi le operazioni logiche e le procedure statistiche ne sono un esempio lampante – erano "progettati principalmente per risolvere problemi preformulati o per elaborare dati secondo procedure predeterminate. L'andamento del calcolo può dipendere dai risultati ottenuti durante l'elaborazione, ma tutte le alternative devono essere previste in anticipo. (Se si presenta

⁶²⁷ "Man-computer symbiosis is an expected development in cooperative interaction between men and electronic computers. It will involve very close coupling between the human and the electronic members of the partnership. The main aims are 1) to let computers facilitate formulative thinking as they now facilitate the solution of formulated problems, and 2) to enable men and computers to cooperate in making decisions and controlling complex situations without inflexible dependence on predetermined programs. In the anticipated symbiotic partnership, men will set the goals, formulate the hypotheses, determine the criteria, and perform the evaluations. Computing machines will do the routinizable work that must be done to prepare the way for insights and decisions in technical and scientific thinking".

⁶²⁸ "Such [...] are not symbiotic systems. They are "semi-automatic" systems, systems that started out to be fully automatic but fell short of the goal".

⁶²⁹ "These systems certainly did not consist of "dissimilar organisms living together..." There was only one kind of organism-man-and the rest was there only to help him".

un'alternativa imprevista, l'intero processo si blocca e attende l'estensione necessaria del programma" [Licklider, 1960 trad. mia]⁶³⁰. In questo caso la simbiosi non è necessaria.

Tuttavia, secondo Licklider, non tutti i problemi sono pensabili, formulabili o risolvibili chiaramente in anticipo. "Sarebbe più facile risolverli, e si potrebbero risolvere più velocemente, attraverso una procedura intuitiva di tentativi ed errori in cui il calcolatore collaborasse, individuando falle nel ragionamento o rivelando svolte inaspettate nella soluzione. Altri problemi, semplicemente, non possono essere formulati senza l'aiuto di una macchina di calcolo" [Licklider, 1960 trad. mia]⁶³¹.

È proprio questo l'obiettivo della simbiosi. Portare la macchina all'interno della formulazione dei problemi. Licklider ha idee chiare su come realizzarlo:

[Lo scopo] è portare le macchine calcolatrici effettivamente dentro i processi di pensiero, cosa che deve avvenire in «tempo reale», tempo che scorre troppo velocemente per permettere di usare i computer in modi convenzionali. [...] Pensare in interazione con il computer nello stesso modo in cui pensi insieme con un collega le cui competenze completano le tue richiederà un collegamento più stretto tra uomo e macchina di quello [...] che è possibile oggi [Licklider, 1960 trad. it. in Numerico et al., 2010 p. 39].

Come rendere un calcolatore un "collega"? Come renderlo un altro organismo in simbiosi con quello umano-biologico? Licklider individua nel suo articolo i punti sulla realizzazione di questo programma: "I prerequisiti per il raggiungimento di un'associazione cooperativa ed efficace comprendono sviluppi nella condivisione del tempo di calcolo, nei componenti di memoria, nell'organizzazione della memoria, nei linguaggi di programmazione e nelle apparecchiature di input e output" [Licklider, 1960 trad. mia]⁶³².

⁶³⁰ "Present-day computers are designed primarily to solve preformulated problems or to process data according to predetermined procedures. The course of the computation may be conditional upon results obtained during the computation, but all the alternatives must be foreseen in advance. (If an unforeseen alternative arises, the whole process comes to a halt and awaits the necessary extension of the program.)".

⁶³¹ "They would be easier to solve, and they could be solved faster, through an intuitively guided trial-and-error procedure in which the computer cooperated, turning up flaws in the reasoning or revealing unexpected turns in the solution. Other problems simply cannot be formulated without computing-machine aid".

⁶³² "Prerequisites for the achievement of the effective, cooperative association include developments in computer time sharing, in memory components, in memory organization, in programming languages, and in input and output equipment". Licklider descrive in *Man-Computer Symbiosis* alcuni punti – che diverranno poi programmi di ricerca finanziati dall'IPTO. Egli richiama l'attenzione innanzitutto sulla necessità di una diversa organizzazione del tempo di calcolo, capace di trasformare i calcolatori da strumenti usati in maniera sequenziale a risorse condivise da più utenti, concepiti come veri e propri "centri di pensiero" interconnessi. Centrale è poi il progresso nei componenti di memoria, che dovranno comprendere memorie permanenti, indelebili o a sola lettura, più compatte ed economiche, e con velocità paragonabile a quella degli elementi logici. Connesso a questo è il problema

Questo approfondimento della metafora cibernetica del calcolatore visto come organismo, conduce – come vedremo più avanti nel caso dello stesso von Neumann – a un superamento del “paradigma EDVAC”:

La simbiosi, infatti, propone una visione dei computer nella quale si fondono i loro diversi compiti di calcolo e gestione dei dati con quelli di immagazzinamento e ricerca delle informazioni. Una volta costruito questo legame tra compiti tanto diversi e dopo aver integrato la componente della comunicazione a tutti i livelli (tra operatore e macchina, tra macchine e tra uomini attraverso le macchine, come prevedeva Wiener) siamo di fronte a una macchina nuova rispetto a quella progettata nel 1945 da von Neumann, sebbene sia ancora governata da un processore, una memoria e un insieme di istruzioni di programmazione [Numerico et al., 2010 p.40].

Dal punto di vista dell’implementazione le idee di Licklider per realizzare la sua visione andavano in direzione diversa rispetto a minicomputer come il PDP-8. Egli voleva partire “da un grande mainframe e dividerlo tra molti utenti simultaneamente. Il concetto, noto come *time-sharing* (condivisione temporale), era in contrasto con l’accesso sequenziale tipico dell’elaborazione a lotti (batch)” [Ceruzzi, 2012, p.76-77 trad. mia]⁶³³. Robert Fano, computer scientist del MIT, già collaboratore di Shannon e dello stesso Licklider, descrive così, nel 1966, i vantaggi – e le speranze – promessi dal *time-sharing*:

A un primo sguardo, la condivisione del tempo di calcolo sembra semplicemente una comodità: un mezzo per consentire a più persone di fare un uso più completo della macchina e di risparmiare tempo agli utenti. In pratica, tuttavia, gli esperimenti con questa tecnica hanno dimostrato una vasta gamma di possibilità più interessanti. Essa permette all’utente di condurre un dialogo continuo con la macchina e, di fatto, rende il calcolatore il suo assistente intellettuale. Inoltre, il sistema rende possibile agli utenti di intrattenere un discorso tra loro attraverso la macchina, attingendo al suo vasto deposito di conoscenze e alla sua velocità di calcolo mentre lo fanno. Il sistema di calcolo con condivisione del tempo può unire un gruppo di ricercatori in una ricerca cooperativa per la soluzione di un problema

dell’organizzazione della memoria, che deve permettere un recupero rapido sia per nome sia per schema, superando la ricerca seriale tradizionale: Sul piano dei linguaggi di programmazione, Licklider cita i progressi resi possibili da FORTRAN e dall’ALGOL, osservando come l’obiettivo debba essere una comunicazione uomo-macchina più naturale, centrata sulla specificazione di obiettivi piuttosto che sulla sequenza di passi elementari. Infine, individua l’urgenza di sviluppare nuove apparecchiature di input e output, che superino i limiti della macchina da scrivere elettrica e consentano una vera interazione con i calcolatori: superfici da scrivania per scrivere e disegnare, display murali per l’attività collaborativa di team, fino al riconoscimento e alla produzione automatica del linguaggio parlato.

⁶³³ “Licklider’s initial idea was to start with a large main frame and share it among many users simultaneously, this concept, known as time-sharing, was in contrast to the sequential access of batch operation”.

comune, oppure può servire come un patrimonio comunitario di conoscenze e competenze a cui chiunque può attingere secondo le proprie necessità. Proiettando il concetto su larga scala, si può concepire una simile struttura come una biblioteca straordinariamente potente al servizio di un'intera comunità - in breve, un'utenza pubblica intellettuale [Fano, Corbató, 1966, p.129 trad. mia]⁶³⁴.

Nel sistema *time-sharing* “se implementato correttamente, gli utenti non si sarebbero accorti che altri stavano usando la macchina nello stesso momento, poiché la velocità del computer era molto più elevata della reattività del cervello umano o della destrezza delle dita [...] Ogni utente avrebbe avuto l'illusione (parola usata intenzionalmente) di avere un potente computer a sua completa disposizione” [Ceruzzi, 2012, p.77 trad. mia]⁶³⁵.

Tuttavia il *time-sharing* – malgrado le speranze riposte in esso come aiuto concreto alla cognizione umana⁶³⁶ - aveva delle difficoltà di implementazione. Una voce di dissenso venne da

⁶³⁴ “At first thought time-sharing seems simply a convenience: a means of allowing fuller use of the machine by more people and of saving time for the users. In practice, however, experiments with the technique have demonstrated a wide range of more interesting possibilities. It enables the user to conduct a continuous dialogue with the machine and in effect makes the computer his intellectual assistant. Further, the system makes it possible for the users to carry on a discourse with one another through the machine, drawing on its large store of knowledge and its computing speed as they do so. The time-sharing computer system can unite a group of investigators in a cooperative search for the solution to a common problem, or it can serve as a community pool of knowledge and skill on which anyone can draw according to his needs. Projecting the concept on a large scale, one can conceive of such a facility as an extraordinarily powerful library serving an entire community - in short, an intellectual public utility”. È da notare come Fano qui riconosca la paternità del concetto a C. Strachey, che abbiamo già incontrato come collaboratore di Turing sulla macchina di Manchester che aveva descritto nel 1959 con il suo *Time sharing in large, fast computers* [Strachey, 1959]. Fano e Licklider furono protagonisti della nascita del Project MAC al MIT, lanciato nel 1963 con un finanziamento DARPA per esplorare la *machine-aided cognition*. Il nome MAC ebbe fin dall'inizio una duplicità di significati - *Mathematics and Computation, Multiple Access Computer, Man and Computer* - a sottolineare l'idea dell'accesso multiplo e della cooperazione uomo-macchina. Diretto da Robert Fano, con Licklider program manager alla DARPA, il progetto si affermò come uno dei centri più influenti per lo studio dei sistemi operativi, dell'intelligenza artificiale e della teoria della computazione. Da ARPA, Licklider organizzò anche il sostegno dell'Office of Naval Research, che finanziò la proposta del MIT con un contratto iniziale di circa 2,2 milioni di dollari: segno evidente che il time-sharing e la sua implementazione interessavano direttamente il governo e i suoi enti di ricerca. In questo contesto prese forma la collaborazione con General Electric e Bell Labs per lo sviluppo di *Multics*, ambizioso sistema operativo di nuova generazione. Il rifiuto da parte del MIT di adottare l'IBM System/360 spinse IBM a correre ai ripari, adattando l'architettura con il Model 67 e promuovendo OS/TSS (Time Sharing System), senza tuttavia riuscire a fare del 360 una piattaforma nativamente concepita per il time-sharing. [Cfr. Haigh, Ceruzzi, 2021, pp.119-120; Ceruzzi, 2012, p.76].

⁶³⁵ “If it was executed properly, users would be unaware that others were also using the machine because the computer's speeds were far higher than the response of the human brain or the dexterity of human fingers [...] Each user would have the illusion (a word used deliberately) that he or she had a powerful computer at his personal beck and call”.

⁶³⁶ Il “capostipite” di questa linea di pensiero di integrazione con la macchina, come riconosciuto esplicitamente dallo stesso Licklider nella prefazione di *Libraries of the Future* [Licklider, 1965, pp. xii-xiii], non può che essere Vannevar Bush. Le riflessioni di Bush sul ruolo dello scienziato e sull'organizzazione della conoscenza, avviate già negli anni '30 ma inizialmente tenute riservate [cfr. Burke, 1994, pp. 9-10], culminarono pubblicamente nell'articolo *As We May Think* [Bush, 1945], apparso sul numero di luglio di *The Atlantic Monthly*. In esso Bush affrontava il problema dell'*information overload* e della frammentazione specialistica, ponendo come compito del dopoguerra quello di riconvertire l'enorme sforzo interdisciplinare della ricerca in strumenti per la pace. La scienza, osservava, aveva già costruito un “registro della conoscenza” sempre più ampio e specializzato, ma al tempo stesso sempre meno accessibile. La speranza risiedeva nelle nuove tecnologie - fotografia, microfilm,

Wesley A. Clark (1927–2016), collaboratore di Ken Olsen ai Lincoln Laboratories, quindi vicino ai fondatori della DEC. Clark riteneva sbagliato l'approccio time-sharing. Egli riteneva più promettente sviluppare piccoli calcolatori personali utilizzabili dai singoli utenti, piuttosto che dividere un'unica grande macchina. Quest'idea fu respinta da informatici e ingegneri, "ma trovò un pubblico più ricettivo tra i ricercatori in ambito medico, che riconoscevano l'utilità di un piccolo computer da collocare accanto alle attrezzature specializzate nei loro laboratori" [Ceruzzi, 2012, p.77 trad. mia]⁶³⁷.

Con il sostegno dei *National Institutes of Health* costruì nel 1962 il LINC, un computer interattivo concepito per laboratori di ricerca biomedica, che anticipava l'idea di personal computer. Con le parole del suo ideatore il LINC (da Lincoln Lab del MIT) rappresentava

Uno dei primi tentativi di trasformare il calcolatore a programma memorizzato in uno strumento generale per l'uso nei laboratori. In una scelta consapevole di distacco rispetto alla tecnologia del time-sharing [...] LINC fu progettato per essere utilizzato da singoli sperimentatori e anticipò così

registrazione vocale, calcolo elettrico - capaci di rendere questo registro dinamico e consultabile, fino a immaginare, in modo evocativo, un laboratorio del futuro in cui lo scienziato, mentre lavora, possa fotografare, commentare, registrare osservazioni e persino automatizzare procedure matematiche ripetitive, liberando così le proprie energie intellettuali per i compiti più creativi. Al culmine di questa riflessione, Bush ipotizza una macchina mai realizzata, che chiama *Memex*: un grande tavolo da lavoro, simile a una scrivania, dotato di schermi inclinati su cui proiettare testi e immagini, leve e pulsanti per navigare tra i contenuti, e un sistema a microfilm per archiviare e richiamare rapidamente libri, articoli, appunti o fotografie. L'utente poteva persino aggiungere materiali propri, appoggiandoli su una lastra trasparente che li trasferiva automaticamente nel sistema. Pensato come un'estensione artificiale della memoria individuale, il *Memex* rappresentava una sorta di biblioteca personale compatta e dinamica, concepita per garantire un accesso immediato e flessibile alla conoscenza. Al di là della tecnologia adottata, il progetto del *Memex*, tratteggiato nell'articolo del 1945, rappresentava un vero e proprio cambio di paradigma rispetto alle modalità tradizionali di accesso, reperimento, creazione e rappresentazione dell'informazione, e di conseguenza rispetto alla gestione stessa della conoscenza. Il passo cruciale, riguarda la critica ai sistemi di indicizzazione rigidi e artificiali, come ad esempio quello Dewey: "La nostra incapacità di trovare documenti è in gran parte causata dall'artificialità dei sistemi di indicizzazione. Quando i dati di ogni tipo sono memorizzati, vengono organizzati alfabeticamente o numericamente e l'informazione è trovata (quando lo è) inseguendola da una sottoclasse a un'altra sottoclasse. [...] La mente umana non funziona così. Opera per associazioni. Con un argomento in suo possesso, passa istantaneamente al successivo suggerito dall'associazione di pensieri, in accordo con alcune intricate reti [web] di piste [trails] contenute dalle cellule del cervello" [Bush, 1945, trad. it in Numerico et al, 2010, p.27]. L'idea di simulare questa strategia associativa, propria della memoria quando seleziona una sequenza di pensieri, forniva una prospettiva inedita per la gestione delle informazioni. In termini filosofici, richiamava, sebbene Bush citasse Leibniz come capostipite del suo progetto, implicitamente la tradizione empirista di Locke e Hume, secondo cui l'associazione è il primo e più efficace metodo per la formazione delle idee. Pur senza attribuire alla macchina la capacità di emulare pienamente la memoria umana, Bush era convinto che essa potesse potenziarne la facoltà naturale di stabilire connessioni, aumentando così il potere dello studioso di orientarsi nella complessità del sapere. [Bush, 1945, Numerico et. al, 2010 pp. 25-29]-Sul *Memex*, la sua *legacy* e le sue conseguenze nella nascita degli ipertesti è imprescindibile ed. Nyce, Kahn, 1991. Castellucci, 2000, analizza il ruolo della macchina ipotetica di Bush nella nascita di Internet anche in relazione ad altri pionieri del Web come Ted Nelson (1937- vivente) e l'inventore del "www" Tim Berners-Lee.

⁶³⁷ "He was rebuffed by electrical and computer engineers but found a more willing audience among medical researchers, who saw the value of a small computer that would not be out of place among the specialized equipment in their laboratories".

caratteristiche tipiche del moderno personal computer e della workstation personale [...] Naturalmente, il LINC al suo tempo non era né un personal computer né una workstation personale, ma ne prefigurava entrambi [Clark, 1986 trad. mia]⁶³⁸.

Parallelamente, i colleghi di Licklider alla BBN sperimentavano la condivisione su scala ridotta, configurando il PDP-1 per servire più utenti contemporaneamente: un prototipo di uso cooperativo che mostrava le potenzialità del concetto, ma anche i suoi limiti quando si cercava di trasferirlo ai grandi mainframe con decine o centinaia di utenti dotati di strumenti meno potenti. Solo dopo anni e con enormi investimenti furono disponibili sistemi commerciali di time-sharing, ma intanto l'impronta lasciata dalle visioni di Licklider aveva già trasformato l'orizzonte della ricerca informatica, in una via che considerando il computer come uno *strumento di comunicazione* [Licklider, Taylor, 1968; Numerico et. al, 2010, pp.48-51] condurrà a Internet⁶³⁹ [Cfr. Ceruzzi, 2012, p.78].

Lo sviluppo tecnologico della computazione personalizzata dovette tuttavia attendere almeno un decennio e alcune innovazioni decisive: "La drastica riduzione del costo del calcolo, resa possibile dai progressi dell'elettronica a stato solido, rese questa possibilità praticabile e

⁶³⁸ "The LINC represents one of the earliest attempts to put the stored program computer into the form of a general instrument for laboratory use. In a deliberate departure from the technology of Timesharing [...] the LINC was designed for use by individual experimenters and thus anticipated features of the modern personal computer and personal workstation [...] Of course the LINC in its day was neither personal computer nor personal workstation, but it anticipated both". A Clark, 1986 si rimanda per la storia, il design e il funzionamento di questo strumento pionieristico.

⁶³⁹ L'eredità delle idee di Licklider intorno al time-sharing e all'interazione uomo-calcolatore, e del suo impegno per metterle in pratica attraverso l'azione di governo in ARPA, è palpabile nell'antesignana di Internet: ARPANET. Nata nel 1969 con il collegamento dei primi calcolatori in centri di ricerca americani, la rete si estese rapidamente – nel 1971 erano già quindici nodi, in gran parte DEC e nel 1972 fu mostrata al pubblico in una dimostrazione a Washington. Il principio tecnico che la rese possibile fu la commutazione a pacchetto: i dati venivano suddivisi in piccoli blocchi autonomi, ciascuno indirizzato e trasmesso separatamente, anche seguendo percorsi diversi, e ricomposti a destinazione. Una tecnica radicalmente diversa dalle linee telefoniche a circuito, difesa da ARPA contro lo scetticismo di ingegneri e aziende come AT&T. Il 29 ottobre 1969, nell'esperimento che inaugurò ARPANET, lo studente Charley Kline all'UCLA tentò di inviare il comando "LOGIN" a un computer dello Stanford Research Institute: le prime due lettere ("L" e "O") arrivarono, ma alla terza il sistema andò in crash. Il primo messaggio della rete fu quindi, ironicamente, un semplice "LO", presto interpretato come un saluto. Pochi minuti dopo la connessione fu ristabilita e il login completato. I protocolli di instradamento e indirizzamento usati da ARPA furono evoluti nel 1983 nello standard TCP/IP (Transmission Control Protocol / Internet Protocol), che separa la gestione del trasporto affidabile dei dati (TCP) dall'instradamento dei pacchetti tra reti diverse (IP). Proprio questa architettura modulare rese la rete scalabile e trasformò ARPANET nella base tecnica del mondo connesso moderno. Per la storia di ARPANET e delle origini di Internet, si rimanda a Abbate, 2000; Beranek, 2000; Haigh, Ceruzzi, 2021, pp. 145-157. Quest'ultimo fornisce anche una spiegazione accessibile dei sistemi tecnici, come la *packet switching*, che strutturano le reti. Una lettura filosofica della nascita del web, iniziata con ARPANET è in Floridi, 1999, pp.56-87. Una efficace sintesi storica che va dalla nascita di ARPANET passando per il World Wide Web sino ai *social network* e all'introduzione degli smartphone è in Ceruzzi, 2012, pp.120-154. La vicenda intellettuale e biografica di quello che può essere considerato l'inventore del web, Tim Berners-Lee è narrata dallo stesso in Berners-Lee, 1999. Intorno al lavoro di quest'ultimo si veda anche Numerico et al., 2010 pp.54-60.

fu sfruttata da giovani come Steve Jobs e Bill Gates, che non facevano parte dell'ambiente accademico di ricerca" [Ceruzzi, 2012, p.78 trad. mia]⁶⁴⁰.

La prima di queste innovazioni fu lo sviluppo del circuito integrato (IC), un dispositivo che riuniva su un singolo chip di silicio più transistor e componenti passivi, permettendo di considerare il circuito come un'unità compatta e non come un insieme di elementi discreti. "I circuiti integrati (IC), che hanno in gran parte sostituito i circuiti costruiti con transistor discreti, sono essi stessi semplicemente insiemi di transistor e altri componenti realizzati su un singolo chip di materiale semiconduttore" [Horowitz, Hill, 2015, p.71 trad. mia]⁶⁴¹.

L'invenzione fu compiuta quasi simultaneamente nel 1958 da Jack St. Clair Kilby (1923–2005), ingegnere elettrico alla *Texas Instruments*, che realizzò il primo prototipo in germanio [Kilby, 1964, US Patent 3,138,743], e nel 1959 da Robert Norton Noyce (1927–1990), ingegnere alla *Fairchild Semiconductor* che utilizzò il processo planare sviluppato dal suo collega Jean Hoerni: una tecnica che sfruttava, invece, strati di ossido di silicio per isolare i transistor e depositare i collegamenti direttamente sulla superficie del chip, rendendo possibile la produzione di massa e l'affidabilità dei dispositivi [Noyce, 1961, US Patent 2,981,877]. La disputa sui brevetti portò infine a riconoscere il merito a entrambi, e nel 2000 Kilby ottenne il Premio Nobel per la Fisica, che sicuramente avrebbe ottenuto anche Noyce se non fosse venuto a mancare appena prima. [Ceruzzi, 2012, pp. 82-88].

La *Fairchild Semiconductor* aprì i battenti nell'ottobre 1957 al 844 East Charleston Road, al confine tra Palo Alto e Mountain View, Da lì prese avvio la strada della computazione che conduceva in California [Lecuyer et. al, 2010].

L'invenzione del circuito integrato ebbe effetti in due direzioni. La prima fu l'impiego di questa nuova tecnologia in applicazioni governative, scientifiche e militari. Il primo impulso venne dal settore aerospaziale statunitense, che richiedeva sistemi compatti, leggeri e

⁶⁴⁰ "The dramatic lowering of the cost of computing that came with advances in solid-state electronics made that practical, and that was exploited by young people like Steve Jobs and Bill Gates who were not part of the academic research environment".

⁶⁴¹ "The transistor is the essential ingredient of every electronic circuit, from the simplest amplifier or oscillator to the most elaborate digital computer. Integrated circuits (ICs), which have largely replaced circuits constructed from discrete transistors, are themselves merely arrays of transistors and other components built from a single chip of semiconductor material". Anche i prodromi dell'IC vanno individuati nella ricerca bellica, in particolare nella spoletta di prossimità, che richiese tubi a vuoto miniaturizzati e resistenti per funzionare all'interno di un proiettile "grande come un cono gelato". Da quel lavoro nacque anche il circuito stampato: un modo per collegare i componenti depositando materiale conduttivo su una lastra isolante invece che con fili. I discendenti di quei circuiti si trovano oggi in quasi tutti i dispositivi digitali, e l'idea stessa di "stampare" un circuito anticipava l'invenzione dell'IC [Cfr. Ceruzzi, 2012, pp. 81-82].

soprattutto affidabili, come nei programmi per il missile balistico *Minuteman*, sviluppati con il contributo della Texas Instruments [Platzek, Kilby, 1964].

La necessità di ridurre i guasti dovuti a un singolo componente portò all'introduzione della *clean room* [Hollbrook, 2009] dove l'aria veniva filtrata e gli operatori indossavano tute protettive per eliminare le particelle di polvere responsabili di cortocircuiti e difetti nei dispositivi⁶⁴². Questi standard si trasferirono rapidamente alla produzione della componentistica, in cui la resa dei *wafer* di silicio (le sottili lastre da cui si ricavano i circuiti integrati) era costantemente minacciata da impurità microscopiche. A metà degli anni Sessanta, l'Aeronautica acquistò in massa circuiti integrati, seguita dalla NASA per i computer di guida del programma Apollo [Hall, 1996]⁶⁴³. Per un breve periodo di tempo la maggior parte dei circuiti integrati si trovava nei missili o nello spazio [Ceruzzi, 2012, pp.88-90].

La seconda direzione fu quella del mercato privato, che dalle imprese sorte lungo la Route 128 - nei sobborghi di Boston, dove si costruivano calcolatori basati sulla componentistica logica standardizzata introdotta da aziende come Fairchild e Texas Instruments - condusse a quella che il giornalista Don Hoefler (1922-1986) definì per la prima volta, nel 1971, "*Silicon Valley*" [Hoefler, 1971].

Fairchild rimase a lungo l'azienda leader, ma fin dall'inizio del suo impegno nei circuiti integrati molti dei suoi migliori ingegneri lasciarono l'impresa per fondarne di nuove nei dintorni. Si diceva scherzosamente che in Silicon Valley si potesse cambiare lavoro semplicemente sbagliando parcheggio al mattino. Questo flusso continuo di competenze, seppur dirompente nel breve periodo, contribuì in ultima analisi al successo della regione del nord della California come motore di innovazione [Ceruzzi, 2012, p.91-92].

Tra le aziende nate da Fairchild vi fu la *Intel*, fondata nel 1968 da Robert Noyce e Gordon Moore, presto raggiunti da Andrew Grove (1936-2016), quasi ironicamente, quest'ultimo era un altro protagonista della rivoluzione informatica di origine ungherese. Nel 1970 Intel impose sul mercato la memoria 1103, la prima memoria DRAM (*Dynamic Random Access Memory*) capace di immagazzinare circa 1.000 bit (128 byte), segnando l'inizio della sostituzione delle memorie a nucleo magnetico con quelle a semiconduttore [Bruce et al, 2008, p.457].

⁶⁴² Le misure di sicurezza per garantire l'affidabilità erano tali che, durante ogni fase di produzione di ciascun componente del *Minuteman*, veniva tenuto un registro dettagliato con le operazioni svolte e i nomi degli addetti responsabili. In questo modo, se un pezzo falliva anche mesi dopo nei test, era possibile risalire con precisione al lotto di provenienza e, qualora risultasse difettoso, ritirare dal servizio tutti i sistemi che lo impiegavano. Ceruzzi, 2012, pp. 89-90.

⁶⁴³ Si rimanda allo stesso per l'analisi dei sistemi computerizzati del progetto Apollo e del ruolo di questo programma nello sviluppo dell'industria dei semi-conduttori.

L'enfasi sulla memoria non è casuale. Come nota P.Ceruzzi, "indicava un potenziale problema legato all'invenzione stessa: si poteva costruire un circuito complesso su un singolo chip di silicio, ma più il circuito era complesso, più era specializzato nella sua funzione, e quindi più ristretto era il suo mercato" [Ceruzzi, 2012, p.92 trad. mia]⁶⁴⁴.

La questione della memoria non riguarda soltanto un problema tecnologico legato alla produzione di massa. Essa tocca un nodo più profondo, che attraversa la nostra analisi sin dalle riflessioni di Turing: il ruolo della memoria del calcolatore come condizione essenziale per trasferire nelle macchine le facoltà della mente e del cervello umano - l'intelligenza - e per progettare architetture che fossero, sul piano del design e delle capacità, un analogo funzionale delle abilità logiche dell'uomo.

Dal relè alla cibernetica, dalla teoria dell'informazione alla "caccia" alla soluzione delle equazioni differenziali e ai problemi di matematica probabilistica, fino al principio del programma memorizzato e, come vedremo più avanti, alle reti neurali, la storia della computazione si è sempre misurata con questo nodo centrale: la gestione della complessità.

Una delle caratteristiche dell'informatica più sconcertanti per il profano è come capacità tanto sorprendenti possano emergere dalla combinazione di pochissimi circuiti logici fondamentali: i circuiti logici AND, OR, NOT, o i loro equivalenti matematici-somma, moltiplicazione o negazione dei numeri 1 e 0. La risposta è che questi circuiti devono essere aggregati in quantità sufficienti: poche decine per eseguire un'aritmetica semplice, qualche centinaio per calcoli più complessi, decine di migliaia per creare un computer digitale, milioni o miliardi per archiviare e manipolare immagini, e così via. Oltre ai componenti attivi come tubi a vuoto o transistor, un computer necessita anche di molti dispositivi passivi, come resistori, diodi e condensatori. L'elemento chiave della progettazione dei computer, tanto nel software quanto nell'hardware, è la gestione della complessità, dai livelli inferiori dei circuiti logici fino a livelli sempre più alti che si stratificano uno sopra l'altro. Si può paragonare ciò al numero di neuroni nei cervelli animali, dal platelminto, al gatto, fino all'*Homo sapiens*-anche se la storia della ricerca sull'intelligenza artificiale ha dimostrato che paragonare un cervello umano a un computer può tanto distorcere quanto chiarire [Ceruzzi, p.82-83 trad. mia]⁶⁴⁵.

⁶⁴⁴ "Emphasis on a memory chip was no accident and pointed to a potential problem with the invention: one could construct a complex circuit on a single chip of silicon, but the more complex the circuit, the more specialized its function, and hence the narrower its market".

⁶⁴⁵ "One characteristic of computing that is most baffling to the layperson is how such astonishing capabilities can arise from a combination of only a few basic logical circuits: the AND, OR, NOT circuits of logic, or their mathematical equivalents: the addition, multiplication, or negation of the digits 1 and 0. The answer is that these circuits must be aggregated in sufficient numbers: a few dozen to perform simple arithmetic, a few hundred to perform more complex calculations, tens of thousands to create a digital computer, millions or billions to store and manipulate images, and so on. Besides the active components like vacuum tubes or transistors, a computer

Abbiamo visto in precedenza come i sistemi operativi abbiano rappresentato una prima forma di gestione della complessità a livello software, semplificando il rapporto con l'operatore che un tempo doveva programmare con cavi o schede perforate. Una innovazione decisiva giunse anche a livello hardware: il *microprocessore*. “Seconda soltanto all'aeroplano, il microprocessore fu la più grande invenzione del ventesimo secolo” [Ceruzzi, 2012, p. 103 trad. mia]⁶⁴⁶.

Intorno al 1970 gli ingegneri che lavoravano sull'elettronica a semiconduttore iniziarono a rendersi conto che il numero di componenti collocabili su un circuito integrato stava crescendo in modo esponenziale. La *slope* della Legge di Moore cominciava a delinearci con chiarezza. Questo scenario apriva però due problemi: il primo riguardava le dinamiche di specializzazione e universalizzazione della tecnologia, il cosiddetto *commonality problem*, il problema della comunanza, ossia la difficoltà, nel caso della computazione, di produrre chip complessi e al tempo stesso di impiego generale; il secondo toccava invece il cuore stesso del design del computer, sin dal *First Draft Report on the EDVAC* del 1945 di von Neumann: la struttura logica del calcolatore, ispirata a quella biologica e informazionale del cervello e della mente, trovava infatti un corrispettivo diretto nell'architettura della sua implementazione fisica e delle componenti che la realizzavano. In questo contesto i minicomputer come il PDP-8 si stavano affermando rispetto ai mainframe IBM anche per i vantaggi architetturali che offrivano. “Se Intel o un'altra azienda produttrice di semiconduttori avesse costruito un computer a chip singolo, avrebbe minato una delle ragioni d'esistenza di queste aziende” [Ceruzzi, 2012, p.101]⁶⁴⁷.

I produttori di componentistica elettronica integrata avevano trovato una soluzione a questi due problemi. Abbandonate le velleità di costruire *cervelli giganti* – come venivano raffigurati i mainframe – producevano chip capaci di implementare le funzioni logiche fondamentali e di garantire prestazioni dignitose. Non erano in grado, tuttavia, di riprodurre la complessità architetturale modulare dei grandi calcolatori, che permetteva di ottimizzare

also requires many passive devices such as resistors, diodes, and capacitors. The key element of computer design, software as well as hardware, is to manage the complexity from the lower levels of logical circuits to ever-higher levels that nest above one another. One may compare this to the number of neurons in the brains of animals, from the flatworm, to a cat, to Homo sapiens, although the history of artificial intelligence research has shown that comparing a human brain to a computer can distort as much as clarify”.

⁶⁴⁶ “Second only to the airplane, the microprocessor was the greatest invention of the twentieth century”.

⁶⁴⁷ “If Intel or another semiconductor company produced a single-chip computer, it would strip away one reason for those companies' existence”

attività specifiche, un problema che aveva accompagnato la costruzione dei computer sin dall'ENIAC. "I progettisti di circuiti personalizzati potevano ottenere prestazioni migliori, ma la cosa cruciale era che al cliente non importava. Il motivo era che il circuito integrato offriva prestazioni abbastanza buone a un prezzo drasticamente più basso, era più affidabile, più piccolo, e consumava molta meno energia". [Ceruzzi, 2012 p. 101 trad. mia]⁶⁴⁸.

La standardizzazione e l'universalizzazione offerta dagli sviluppi dei circuiti integrati avevano aperto un mercato nuovo: quello della complessità, prima realizzata nelle sofisticate costose apparecchiature che implementavano le funzioni del calcolatore. [Cfr. Moore, 1976]⁶⁴⁹.

In questo contesto Intel annunciò nel 1971 il suo microprocessore, il chip 4004, presentato su una rivista del settore, *Electronic News*, come "computer microprogrammabile su un chip" [*Electronic News*, novembre 1971, trad. mia]⁶⁵⁰. Il merito dell'invenzione del primo microprocessore, l'Intel 4004, "il primo esempio di un circuito complesso a logica casuale realizzato utilizzando la tecnologia MOS [*Metal-Oxide-Semiconductor*, tecnologia basata su transistor a effetto di campo con gate isolato]" [Faggin et al., 1996, trad. mia]⁶⁵¹ è attribuito Marcian ("Ted") Hoff (1937-), Stan Mazor (1941-) e dall'italiano Federico Faggin (1941-), ingegneri della Intel: Hoff ideò l'architettura, Mazor collaborò alla definizione logica, mentre Faggin guidò il progetto e lo sviluppo tecnologico.

Il 4004 e i suoi successori immediati, l'8008 (1972) e l'8080 (1974), ebbero successo perché furono in grado di rispondere alle due obiezioni sopra menzionate che avevano frenato lo sviluppo dei circuiti integrati complessi: quello della *comunanza*, superato grazie all'introduzione di chip di input-output e di memoria che permettevano di adattare l'architettura a molte applicazioni diverse; e le riserve sul design furono vinte offrendo un sistema economico, compatto e a basso consumo. Intel accompagnò i processori con kit di sviluppo composti dal microprocessore, memorie ROM e RAM, alimentazione e periferiche di

⁶⁴⁸ "The chip manufacturers had already faced a simpler version of this argument: they produced chips that performed simple logic functions and gave decent performance, though that performance was inferior to custom-designed" circuits that optimized each individual part. Custom-circuit designers could produce better circuits, but what was crucial was that the customer did not care. The reason was that the IC offered good enough performance at a dramatically lower price, it was more reliable, it was smaller, and it consumed far less power".

⁶⁴⁹ Si rimanda allo stesso [Moore, 1976] per la visione di G. Moore, protagonista di primo piano nello sviluppo dell'elettronica, che da "tecnologo" interpreta il rapporto tra complessità ed evoluzione dei circuiti integrati sino all'introduzione del microprocessore.

⁶⁵⁰ "A micro-programmable computer on a chip".

⁶⁵¹ "The 4004 is the first example of a complex random-logic circuit built using silicon-gate MOS technology." Allo stesso si rimanda per la storia, il design, le prestazioni e i ruoli dei protagonisti nella creazione del primo microprocessore. La vicenda biografica di Federico Faggin in cui viene narrata la storia di questa invenzione rivoluzionaria e il contributo decisivo da lui apportato nello sviluppo della tecnologia *silicon-gate* e nell'ingegnerizzazione del chip, è in Faggin, 2020.

base: erano a tutti gli effetti dei computer general purpose su singolo chip [Faggin 2018; Ceruzzi, 2012, p.103-105; Moore, 1976].

Con l'introduzione del microprocessore il processo di riduzione artificiale del computer umano che abbiamo più volte incontrato - il cui emblema era Wiener costretto negli abiti militari o le donne della Moore School impegnate nei calcoli - si compie ora in pochi centimetri di silicio.

Anche se l'Intel continuava a proporli soltanto come dispositivi per sistemi *embedded* (cioè computer incorporati in apparecchiature più grandi, progettati per svolgere funzioni specifiche, ad esempio in strumenti industriali o macchine di controllo), i microchip, e i loro kit, non furono così interpretati da miriadi di hobbisti, radioamatori e computer scientist dilettanti.

"Quando la Micro Instrumentation and Telemetry Systems (MITS) annunciò il suo kit Altair sulla copertina del numero di gennaio 1975 di *Popular Electronics*, si aprirono le dighe" [Ceruzzi, 2012, p.105; Roberts, Yates, 1975, *Popular Electronics*, gennaio 1975]⁶⁵². Il titolo, in blu acceso, sulla copertina di *Popular Electronics* era emblematico della rivoluzione che lo strumento preannunciava: "ALTAIR 8800 / *"The most powerful minicomputer project ever presented - can be built for under \$400"*. [Roberts, Yates, 1975, *Popular Electronics*, gennaio 1975]⁶⁵³.

La nuova gestione della complessità introdotta dal processore implicava una nuova caratteristica "rivoluzionaria" nel design dei calcolatori. Ne cambiava il livello d'astrazione, passando, come intuì Turing con ACE, dal livello dell'hardware a quello del software. "Il suo effetto sul progettista di apparecchiature lo costringe a concentrarsi in larga misura sui programmi software piuttosto che, come in passato, sull'interconnessione delle porte logiche" [Moore, 1976 trad. mia]⁶⁵⁴. E fu proprio quello che successe quando Paul Allen, un giovane ingegnere alla Honeywell, e Bill Gates, studente universitario ad Harvard, lessero l'articolo su *Popular Electronics*, entrambi lasciarono Cambridge per fondare la Microsoft, un'azienda nata per la scrittura di software per l'Altair [Haigh, Ceruzzi, 2012, p.177]⁶⁵⁵.

⁶⁵² "“When Micro Instrumentation and Telemetry Systems (MITS) announced its ‘Altair’ kit on the cover of the January 1975 issue of *Popular Electronics*, the floodgates opened.”

⁶⁵³ Sull'Altair 8800, per il design, il funzionamento e le caratteristiche si veda Haigh, Ceruzzi, 2021, pp.173-176.

⁶⁵⁴ "The advent of the microprocessor as a general-purpose digital electronic block, whose function is determined by programming, is revolutionary. Its effect on the equipment designer forces him to concern himself in large extent with software programs rather than with the interconnection of logic gates, as before".

⁶⁵⁵ Il linguaggio scelto dai due fondatori della futura Microsoft fu il BASIC (*Beginner's All-purpose Symbolic Instruction Code*), sviluppato nel 1964 da John Kemeny e Thomas Kurtz al Dartmouth College come strumento didattico di facile apprendimento per studenti non specialisti. A lungo disprezzato dagli informatici accademici come un linguaggio "giocattolo", il BASIC doveva però la sua fortuna a una caratteristica decisiva: la semplicità della sintassi interattiva (*PRINT, INPUT, GOTO*), che permetteva di scrivere ed eseguire rapidamente programmi

Era nata una nuova fase della storia della computazione: l'era del personal computer. Nella Silicon Valley nacquero altri computer personali divenuti poi leggendari, basati su processori economici come il 6502 [Carbone, 1979; *Microcomputer Digest*, agosto 1975] come il Commodore PET e il suo successore, il Commodore 64 [Haigh, Ceruzzi, 2021, p.199]. Questi strumenti offrivano una potenzialità impensabile ai creatori dei primi mainframe, la possibilità di giocare ai video-games [Cfr. Haigh, Ceruzzi, 2021, pp. 201-203].

Nel 1976 Steve Wozniak e Steve Jobs presentarono l'Apple I, seguito l'anno successivo dall'Apple II, che si impose come uno dei primi grandi successi commerciali dell'informatica personale. Tutti questi sviluppi erano legati all'ambiente "controculturale", del quale abbiamo visto il retroterra cibernetico nelle pagine precedenti e che qui incontrava la dimensione dell'implementazione resa possibile dallo sviluppo dell'elettronica. "Nella Silicon Valley, giovani appassionati di computer, molti dei quali figli di ingegneri impiegati in aziende locali di elettronica o difesa, formarono l'*Homebrew Computer Club* e si incontravano regolarmente per condividere idee e progetti individuali di computer personali" [Ceruzzi, 2102, p. 112 trad. mia]⁶⁵⁶.

L'espansione della mente promossa per via cibernetica da esponenti della controcultura come Stewart Brand aveva trovato un veicolo concreto nel microprocessore e nella sua diffusione. A questo *milieu* controculturale, libertario e amatoriale appartenevano gli stessi Wozniak e Jobs, insieme a molti altri protagonisti della rivoluzione del personal computer [Markoff, 2005].

Non è possibile, in questo lavoro, seguire oltre la storia della computazione personalizzata fino ai giorni nostri. È però possibile sottolinearne alcune conseguenze, che mostrano il carattere irriducibilmente socio-tecnico delle tecnologie analizzate, le quali acquistano ora maggiore concretezza alla luce dell'esame della loro componente materiale-tecnologica, in una dinamica di reciproca co-determinazione tra dimensione sociale e dimensione tecnica.

anche su macchine con risorse minime. Nella leggendaria "*Open Letter to Hobbyists*" [Gates, 1976] si manifestava uno dei primi segnali di un fenomeno destinato a segnare il mondo dell'informatica: la duplicazione impropria di linguaggi e programmi. Gates denunciava il fatto che il suo BASIC venisse copiato e diffuso illegalmente tramite la duplicazione dei nastri cartacei, ricordando che erano stati spesi oltre 40.000 dollari di tempo-macchina per svilupparlo. Senza un adeguato ritorno economico, sosteneva, lui e i suoi collaboratori non avrebbero avuto incentivi a produrre nuovo software per i personal computer.

⁶⁵⁶ "In Silicon Valley, young computer enthusiasts, many of them children of engineers working at local electronics or defense firms, formed the Homebrew Computer Club and met regularly to share ideas and individual designs for personal computers".

4. Una società omeostatica, una liberazione cibernetica

Dalla cibernetica e dalla teoria dell'informazione, fino alla macchina universale di Turing e all'architettura di von Neumann, si consolidò una retorica che vedeva nel computer un modello funzionale della mente e un principio sistemico capace di riorganizzare la società. Nato dalla guerra e plasmato dalla bomba, il calcolatore assunse negli anni '50 e '60 la sua forma di strumento insostituibile, diffondendosi rapidamente nella scienza e nella società. Questa espansione si incardinò nella scienza e nella società attraverso le *trading zones* e la lingua franca delle simulazioni. L'implementazione del metodo Monte Carlo sull'ENIAC segnò simbolicamente l'apertura di uno spazio digitale in cui il calcolatore appariva non più soltanto tecnologia, ma ambiente cognitivo e sociale che era possibile controllare mediante la retorica sistemica della cibernetica e computare attraverso gli sviluppi del calcolatore seriale. Da tale terreno prese forma, per continuità e reazione, l'idea della computazione personalizzata, che avrebbe condotto alla stagione dell'informatica individuale.

Da questo punto di vista, diventa inevitabile interrogarsi su come considerare il computer e le tecnologie digitali non soltanto come esiti di un determinismo tecnologico, ma come veri e propri artefatti sociotecnici. In questo quadro, non possiamo non considerare il calcolatore da un punto di vista affine a quello degli studi di *Science and Technology Studies* (STS). Secondo gli STS, “conoscenze e artefatti sono prodotti umani, segnati dalle circostanze della loro produzione”. Scienza e tecnologie non sono “generi naturali”, non esiste un metodo scientifico né uno tecnologico privilegiato, e sono ambiti in cui il “lavoro retorico è cruciale”. Vi è una dimensione storica e sociale irriducibile in esse: “sono sociali nel senso che scienziati e ingegneri sono sempre membri di comunità, formati alle pratiche di quelle comunità e necessariamente operanti al loro interno. Queste comunità stabiliscono gli standard dell'indagine e valutano le pretese di conoscenza” [Sismondo, 2010, pp.10-11 trad. mia]⁶⁵⁷.

Il computer non è nato secondo un'evoluzione progressiva e lineare alimentata da una sorta di logica interna, portato avanti in isolamento da personalità scientifiche eccezionali che avevano a che fare solo con fenomeni da studiare e manipolare.

⁶⁵⁷ “Science and Technology Studies (STS) starts from an assumption that science and technology are thoroughly social activities. They are social in that scientists and engineers are always members of communities, trained into the practices of those communities and necessarily working within them. These communities set standards for inquiry and evaluate knowledge claims; In addition, science and technology are arenas in which rhetorical work is crucial: Neither science nor technology is a natural kind, having simple properties that define it once and for all. The sources of knowledge and artifacts are complex and various: there is no privileged scientific method that can translate nature into knowledge, and no technological method that can translate knowledge into artifacts”.

Gli studiosi della STS rifiutano questa affermazione. Essi sostengono che lo sviluppo scientifico e tecnologico è tutt'altro che isolato e non segue un ordine di sviluppo predeterminato o "naturale". Il carattere e la direzione dello sviluppo tecnologico sono influenzati da un'ampia gamma di fattori sociali, tra cui: le decisioni di un'agenzia governativa di finanziare certi tipi di ricerca; eventi sociali come una guerra o un attacco terroristico che stimolano l'interesse e l'impegno nella produzione di certi dispositivi (ad esempio per la sicurezza); forze di mercato che alimentano lo sviluppo in alcuni settori e lo arrestano in altri; l'ambiente giuridico, che può limitare l'innovazione in certi ambiti e favorirla in altri; e sensibilità culturali che attribuiscono significati riprovevoli ad alcune tecnologie e desiderabili ad altre [Johnson, Miller, 2009, pp.13-14 trad. mia]⁶⁵⁸.

Il calcolatore è un artefatto *sociotecnico*, una "combinazione di cose e persone" [Johnson, Miller, 2009, pp.16-17 trad. mia]⁶⁵⁹. Da questo punto di vista, devono essere considerate le configurazioni sociali che lo hanno plasmato e che ha contribuito ad alimentare e a dischiudere come sviluppo tecnologico.

La trasformazione del computer da una macchina grande quanto una stanza a un dispositivo personale da tenere in mano è un paradosso. Da un lato, fu il risultato diretto dei progressi dell'elettronica a stato solido seguiti all'invenzione del transistor nel 1947 e, in quanto tale, un'illustrazione del determinismo tecnologico: il cambiamento sociale guidato dalla tecnologia. Dall'altro lato, l'informatica personale non fu prevista dagli ingegneri informatici; fu piuttosto spinta da visioni idealistiche della controcultura degli anni '60. In questo senso, l'informatica personale fu l'antitesi del determinismo tecnologico [Ceruzzi, 2012, pp.111-112 trad. mia]⁶⁶⁰.

⁶⁵⁸ "STS scholars reject this claim. They argue that scientific and technological development is far from isolated and does not follow a predetermined or "natural" order of development. The character and direction of technological development are influenced by a wide range of social factors including: the decisions a government agency makes to fund certain kinds of research; social incidents such as a war or terrorist attack that spark interest and effort to produce particular kinds of devices (e.g. for security); market forces that feed development in some areas and bring others to a halt; the legal environment, which may constrain innovation in certain areas and facilitate it in others; and cultural sensibilities that lead to objectionable meanings associated with certain technologies and desirable meanings for others".

⁶⁵⁹ "STS scholars reject the idea that technology is material objects, and entreat us always to think of technology as sociotechnical systems (combinations of things and people)".

⁶⁶⁰ "The transformation of the computer from a room sized ensemble of machinery to a handheld personal device is a paradox. On the one hand, it was the direct result of advances in solid-state electronics following the invention of the transistor in 1947 and, as such, an illustration of technological determinism: the driving of social change by technology. On the other hand, personal computing was unanticipated by computer engineers; instead, it was driven by idealistic visions of the 1960s era counterculture. By that measure, personal computing was the antithesis of technological determinism"

In questo paradosso è possibile notare, a livello dell'hardware, la stessa tensione contraddittoria evidenziata in apertura sul piano socio-culturale: tra i timori che l'utopia generata dall'introduzione del calcolatore si trasformasse in distopia, riducendo l'umano a scheda perforata da computare, e le promesse di liberazione, rese possibili dalla comune natura informazionale di uomo e ambiente, individuata nelle macchine stesse.

Alla base di entrambe vi è la retorica della cibernetica. Questa si basava su "un'analogia potente" e, cioè, che "i principi delle macchine a retroazione [*feedback*] informazionale, come il funzionamento di un termostato che controlla una caldaia domestica, potevano essere usati anche per spiegare il comportamento di tutti gli esseri viventi nel loro interagire con l'ambiente", e questo "dal livello della cellula fino a quello della società" [Kline, 2015, pp.1-2 trad. mia]⁶⁶¹.

Wiener aveva reso celebri le metafore del cervello come un computer digitale e della società come un sistema di comunicazione in *Cybernetics* del 1948 e *the Human Use of Human Beings* 1950 [Kline, 2015, p.44-45].

Un momento chiave nella diffusione interdisciplinare delle idee cibernetiche, e nella formazione di una *trading zone* e di un linguaggio comune, furono le Macy Conferences che, nel loro composito parterre di relatori, includevano anche numerosi scienziati sociali, come i già citati Gregory Bateson e Margaret Mead.

È Wiener a riconoscere, nell'introduzione di *Cybernetics* come "uno degli orientamenti di lavoro · emersi dalle idee dibattute ai convegni della Macy [abbia] riguarda[to] l'importanza del concetto e della tecnica di comunicazione nel sistema sociale". Ma subito dopo è lo stesso matematico del MIT a esprimere scetticismo sull'applicazione della cibernetica, e dei suoi fondamenti come teoria dell'informazione, alla società. Afferma Wiener: "I principali fattori che influenzano la società non sono soltanto statistici, e le serie statistiche su cui sono basati sono inoltre eccessivamente brevi [...] Per avere una buona statistica della società, abbiamo bisogno di lunghe serie di dati sotto condizioni essenzialmente costanti", questo malgrado sia "certamente vero che il sistema sociale è un'organizzazione simile a quella individuale, tenuta assieme da un sistema di comunicazioni e caratterizzata da una dinamica in cui processi circolari a feedback svolgono un'importante funzione" [Wiener, 1948, p.49 trad. it.].

⁶⁶¹ "The premise of cybernetics was a powerful analogy: that the principles of information-feedback machines, which explained how a thermostat controlled a household furnace, for example, could also explain how all living things-from the level of the cell to that of society-behaved as they interacted with their environment".

Tuttavia, malgrado le riserve di Wiener, la cibernetica ebbe una forte influenza nelle scienze sociali. Come nota il fisico e storico della scienza Steve J. Heims (il primo a ricostruire storicamente le conferenze Macy)

Le idee della cibernetica si prestano a essere utilizzate nei contesti ideologici più diversi. Alcuni le hanno usate come strumento per favorire la centralizzazione del potere, Wiener, al contrario, le usò per argomentare appassionatamente contro la concentrazione del potere politico e amministrativo e per esaltare i pregi delle piccole comunità interattive. Nei paesi comunisti i filosofi sociali adottarono la cibernetica, dopo averla in un primo tempo rifiutata, giudicandola perfettamente coerente con il materialismo dialettico e particolarmente utile nella descrizione di modelli sociali e politici che potessero conciliare il decentramento del controllo con il mantenimento di funzioni generali centralizzate. I sociologi americani l'hanno utilizzata nell'elaborazione di un'analisi funzionale dello status quo: applicando il linguaggio della cibernetica alla società, non tennero conto di elementi quali il teorema dell'incompletezza di Gödel o i paradossi di Russell, come invece fecero Wiener e Bateson, e i loro modelli descrittivi risultarono così, spesso, artificiosamente rigidi e deterministici [Heims, 1991, p.207 trad. it.].

Tra gli esempi di scienziati sociali a cui è riservato il poco lusinghiero giudizio di Heims, si può prendere come esempio, come ha fatto lo storico della cibernetica R. Kline, il sociologo di Harvard, Talcott Parsons (1902- 1979), "la cui teoria struttural-funzionalista della società (una disciplina sociale "maestra") dominò la sociologia e l'antropologia durante la Guerra Fredda" [Kline, 2015, p.148 trad. mia]⁶⁶².

Parsons propose una teoria dell'azione e della società, articolata come quattro sottosistemi funzionali: l'organismo, che opera al livello fisico e biologico; il sistema sociale, che riguarda l'interazione tra individui e i ruoli istituzionali; il sistema culturale, che organizza i significati simbolici e le norme condivise e il sistema della personalità, che rappresenta la dimensione soggettiva e motivazionale dell'individuo [Parsons, 1951; Cfr. Kline, 2015, p.148]. Questi quattro sottosistemi dovevano rimanere insieme mediante meccanismi *omeostatici* di *regolazione e controllo*, cioè i concetti chiave della cibernetica. Parson stesso ricorda:

Il chiarimento del problema del controllo è stato enormemente favorito dall'emergere, in un momento per me cruciale, di un nuovo sviluppo della scienza generale – vale a dire della cibernetica nel

⁶⁶² "My final example in this section is Harvard sociologist Talcott Parsons, whose structural-functional theory of society (a master social science) dominated sociology and anthropology in the Cold War".

suo stretto rapporto con la teoria dell'informazione. È ora possibile affermare che la forma basilare di controllo nei sistemi attivi sia di tipo cibernetico e non si basi, come è stato generalmente proposto, principalmente sull'analogia con gli aspetti coercitivo-compulsivi dei processi nei quali è coinvolto il potere politico [Parsons, 1970, p.831, p.850 trad. it. in Heims, 1991, p.212 trad. it.].

Questa *anagnōrisis* di Parsons sulle potenzialità omeostatiche e di controllo della cibernetica è da far risalire ai contatti del sociologo americano con i protagonisti alle Macy Conferences. Parsons divenne un "fanatico" della cibernetica. "L'analisi funzionale di Parsons, spesso presentata sia come teoria sociologica della borghesia che come contraltare della sociologia marxista, poneva l'accento su quei meccanismi omeostatici, come il feedback negativo, che mantengono stabili le strutture di fondo di un sistema sociale" [Heims, 1991, p.212 trad. it.].

La metafora cibernetica, diveniva la struttura concettuale del mantenimento dell'ordine sociale, la stessa società che veniva a plasmarsi e razionalizzarsi intorno al calcolatore, il cui sviluppo e il suo impiego venivano riprodotti e giustificati dallo stesso armamentario concettuale. La società di Talcott Parson era quella in cui "era accaduto qualcosa di notevole nel mondo nell'ultima generazione circa. Nonostante l'enorme varietà di valori e di altri elementi culturali, si è giunti a un consenso quasi globale nella valutazione della produttività economica di alto livello industriale. In sostanza, ciò significa l'accettazione generale dei modelli industriali della società occidentale come fornitori di almeno una componente essenziale della "buona società" [Parsons, 1960, p.21]⁶⁶³.

In essa, i conflitti sociali dovevano essere ricondotto ad una situazione di equilibrio omeostatico, usando "circuiti di retroazione cibernetica per modellare il processo decisionale umano e le interazioni sociali nelle scienze del management, nella psicologia, nella sociologia, nelle scienze politiche e nell'antropologia" [Kline, 2017, p.148 trad. mia.]⁶⁶⁴. L'omeostasi, da criterio per determinare il comportamento del vivente e successivamente applicabile anche alle

⁶⁶³ "[A] remarkable thing has happened in the world in the last generation or so. In spite of the immense range of variation in values and other elements of culture, . . . there has come to be an almost world-wide consensus on the evaluation of high, industrial-level, economic productivity. . . . Essentially it means the general acceptance of the industrial models of Western society as providing at least *one* essential part of the "good society."

⁶⁶⁴ "Herbert Simon, George Miller, Karl Deutsch, Roman Jakobson, Talcott Parsons, and Gregory Bateson. They employed cybernetics and information theory in two ways to explain human behavior. One method used an "information calculus" to model human judgments and the structure of language in psychology and linguistics. The second method used cybernetic feedback circuits to model human decision making and social interactions in management science, psychology, sociology, political science, and anthropology".

macchine, diveniva principio di organizzazione sociale, in un progressivo sfumarsi delle differenze.

Parsons “applicò” l’intuizione cibernetica secondo cui sistemi ad alto contenuto informativo ma a bassa energia (come il termostato domestico) possono controllare sistemi a basso contenuto informativo ma ad alta energia (come la caldaia domestica). Ad esempio, il sistema culturale ad alta informazione di Parsons poteva controllare il sistema dell’organismo, caratterizzato da bassa energia” [Kline, 2017, p.148 trad. it.].

Ciò significava che, per preservare l’equilibrio - l’omeostasi sociale - era necessario esercitare un controllo sulla produzione di informazione e sulla cultura. Non possono non tornare alla mente, a questo proposito, le proteste all’Università di Berkeley con cui abbiamo aperto questo lavoro, segnate dal timore degli studenti di ridursi a mere *punched-cards*.

Ma tutto questo non si limitava al piano scientifico o accademico. La cibernetica, insieme alla sua retorica, si impose – anche grazie alla posizione di predominanza sul mercato di imprese come l’IBM - come metafora dell’organizzazione della società stessa: un organismo-sistema di secondo ordine, fondato sull’elaborazione dell’informazione, che connette tra loro sottosistemi concepiti anch’essi come entità informazionali. Il progetto SAGE ne fu l’emblema. Non si trattava solo di un sistema tecnico-militare, ma di un organismo informazionale integrato, progettato per proteggere un altro organismo sistemico - la società - intesa, secondo la visione cibernetica, come una rete complessa di raccolta, scambio e controllo dell’informazione. Anche il decisore umano era incluso in questa rete: parte attiva del sistema, elaborava dati che non avrebbe potuto ottenere senza l’intermediazione automatica del calcolatore. L’integrazione tra uomo e macchina diveniva un altro snodo di comunicazione di questo enorme apparato di informazioni⁶⁶⁵.

⁶⁶⁵ Paul N. Edwards mostra come la nascita delle scienze cognitive si intrecciò con contesti e finalità non neutrali. Centrale fu l’introduzione dei concetti cibernetici e della teoria dell’informazione – in primis la riduzione del rumore – nella psicologia, con l’obiettivo di migliorare l’interazione tra uomo e macchina all’interno di sistemi complessi. Le ricerche sviluppate nel dopoguerra – dal *Psycho-Acoustic Laboratory* di Harvard fino ai programmi di difesa aerea – avevano già configurato l’essere umano come canale di comunicazione con capacità limitata, da integrare e ottimizzare dentro sistemi uomo-macchina a finalità militari. In questo quadro, il simposio del settembre 1956 al MIT, in cui le scienze cognitive presero forma, può essere letto come un momento in cui linee di ricerca diverse (memoria, linguaggio, simulazione) furono ricondotte sotto il paradigma comune dell’*“information processing”*. Tecniche come i test di articolazione, le misure di capacità di canale o le regole sul “numero magico sette” [Miller, 1956] rinviavano a un retroterra – cui lo stesso Miller era legato anche attraverso il progetto SAGE – di indagini sul rapporto tra comunicazione, rumore e prestazioni cognitive, che offriva criteri per la progettazione di interfacce e compiti connessi all’uso del calcolatore, la cui centralità andava rapidamente crescendo. L’integrazione tra uomo e macchina come elaboratori d’informazione rispondeva dunque anche all’esigenza di affinare questa relazione con macchine divenute elemento cardine della riorganizzazione economica e militare della società, concepita come un enorme apparato, un vero e proprio “organismo” di elaborazione dell’informazione {Cfr. Edwards, 1996, pp. 207-237}. Affinché ciò fosse possibile, era necessaria una

Lo storico della scienza e della tecnologia Paul N. Edwards, che nei suoi studi ha indagato il nesso tra infrastrutture computazionali, potere e cultura, sottolinea nel suo *The Closed World: Computers and the Politics of Discourse in Cold War America* [Edwards, 1996] come l'interazione tra sistemi organici e inorganici di elaborazione dell'informazione, uniformemente quantificabile in bit, producesse effetti tutt'altro che politicamente o socialmente neutri, risultando invece funzionale all'assetto ideologico e istituzionale della Guerra Fredda.

Sia l'ingegneria centrata sia la politica del discorso del *mondo chiuso* ruotavano attorno ai problemi dell'integrazione uomo-macchina: costruire armi, sistemi e strategie le cui componenti umane e meccaniche potessero funzionare come una rete continua e integrata, anche su scala globale e nei tempi estremamente compressi della guerra nucleare tra superpotenze. In quanto macchine logiche capaci di manipolare simboli, i computer avrebbero automatizzato o assistito compiti di percezione, ragionamento e controllo all'interno di sistemi integrati. Tali obiettivi, realizzati per la prima volta con le armi antiaeree dell'epoca della Seconda guerra mondiale, contribuirono a dar forma sia alla cibernetica - la grande teoria dell'informazione e del controllo nei sistemi biologici e meccanici - sia all'intelligenza artificiale (IA), il software che simula un pensiero simbolico complesso [Edwards, 1996. p.1 trad, mia]⁶⁶⁶.

Di tutt'altro avviso era Wiener, per il quale la computazione personalizzata offriva un supporto "realizzato" a chi temeva o contestava la configurazione sociale che la cibernetica e la teoria dell'informazione - anch'esse, come abbiamo visto, nate dallo sforzo bellico - stavano producendo. Come emerge dalle considerazioni di S. J. Heims sul rapporto tra cibernetica e

teoria capace di tenere insieme sistemi umani e artificiali, organici e inorganici, entro un unico quadro concettuale. Il calcolatore diventava così il perno di una nuova psicologia, che vedeva le menti "concepite come un sistema di risoluzione dei problemi, autocontrollo ed elaborazione simbolica" ["At the same time, computers inspired new psychological theories built around concepts of 'information processing.' Cybernetics, AI, and cognitive psychology relied crucially upon computers as metaphors and models for minds conceived as problem-solving, self-controlling, symbol-processing systems", Edwards, 1996, p. 2, trad. mia]. Queste menti, organizzate in modo seriale, dovevano poter essere messe in relazione con i computer, come il 701 dell'IBM definito dal colosso statunitense "*electronic data processing machine*" [Ceruzzi, 2003, p. 34]. Non solo la cibernetica, dunque, ma anche la nascente psicologia dell'elaborazione dell'informazione risultava strettamente intrecciata con una struttura sociale che si sorreggeva e si plasmava sulla computazione.

⁶⁶⁶ "Both the centered engineering and the politics of closed-world discourse around problems of human-machine integration: building weapons, systems, and strategies whose human and machine components could function as a seamless web, even on the global scales and in the vastly compressed time frames of superpower nuclear war. As symbol-manipulating logic machines, computers would automate or assist tasks of perception, reasoning, and control in integrated systems. Such goals, first accomplished in World War II-era anti-aircraft weapons, helped form both cybernetics, the grand theory of information and control in biological and mechanical systems, and artificial intelligence (AI), software that simulated complex symbolic thought".

sociologia, Wiener aveva infatti una prospettiva diversa riguardo all'impiego dei concetti della *comunicazione* e del *controllo* nell'analisi sociale.

“Per Wiener il nemico principale non era mai la Russia, bensì l’uso *disumano degli esseri umani*, e questo includeva per lui lo sfruttamento, l'imposizione di rigidità e l'assenza di feedback e di una comunicazione onesta a due vie nell'organizzazione sociale” [Heims, 1980, p.311 enfasi mia]⁶⁶⁷. Tra i pericoli il padre della cibernetica annoverava quelli di un governo-stato tecnocratico in cui

I leader politici possano tentare di controllare le loro popolazioni non attraverso le macchine in sé, ma mediante tecniche politiche tanto ristrette e indifferenti alle possibilità umane come se fossero state, in effetti, concepite meccanicamente. La grande debolezza della macchina – la debolezza che finora ci ha salvati dall'essere da essa dominati – è che non può ancora tener conto dell'enorme gamma di probabilità che caratterizza la condizione umana. Il dominio della macchina presuppone una società negli stadi finali di un crescente stato entropico, in cui la probabilità è trascurabile e le differenze statistiche tra individui sono nulle. [Wiener, 1950, p.181 trad. mia]⁶⁶⁸

⁶⁶⁷ “To Wiener the primary enemy was never Russia but rather the inhuman use of human beings, and for him this included exploitation, imposition of rigidity, and absence of feedback and honest two-way communication in the social organization”.

⁶⁶⁸ “Its real danger, however, is the quite different one that such machines, though helpless by themselves, may be used by a human being or a block of human beings to increase their control over the rest of the human race or that political leaders may attempt to control their populations by means not of machines themselves but through political techniques as narrow and indifferent to human possibility as if they had, in fact, been conceived mechanically. The great weakness of the machine—the weakness that saves us so far from being dominated by it—is that it cannot yet take into account the vast range of probability that characterizes the human situation. The dominance of the machine presupposes a society in the last stages of increasing entropy, where probability is negligible and where the statistical differences among individuals are nil.” L’obiettivo polemico di Wiener, qui, è l'impiego della teoria dei giochi di von Neumann, che riduceva le possibilità di apprendimento e di comportamento dell'essere umano al solo criterio di *minimax* calcolabile dalle macchine (che egli esemplifica con il comportamento del giocatore scacchistico meccanizzato che abbiamo già incontrato in Shannon) e che il padre della cibernetica vedeva ormai diffondersi come criterio generalizzato di organizzazione della società – e dei rapporti geopolitici – a tutti i livelli. “Già oggi la concezione della guerra che ispira alcuni dei nostri nuovi uffici statali, che stanno sviluppando le conseguenze della teoria dei giochi di von Neumann, è sufficientemente estesa da comprendere tutte le attività civili durante la guerra, prima della guerra, e forse perfino tra una guerra e l'altra” [Wiener, 1950, p.225 trad. it.; Cfr. Wiener 1950 pp.221-226 trad. it]. Non è stato possibile approfondire qui questo aspetto del pensiero di von Neumann, ma si rimanda a Israel, Millán Gasca, 2008 181-194 e Leonard, 2010, pp. 293-343. A riprova dell'avversione di Wiener per l'applicazione della teoria dei giochi, e della sua “filosofia” di fondo alla sfera sociale possiamo leggere in *Cibernetica*: “Naturalmente, l'immagine del giocatore di von Neumann, come persona completamente razionale e completamente spietata, è un'astrazione e una alterazione dei fatti”. Nei criteri del *minimax* a livello dei sistemi sociali non si verifica alcuna omeostasi, al contrario essi sono fortemente *entropici*. Prosegue infatti Wiener: “I giocatori individuali sono sospinti dalla loro cupidigia a formare coalizioni; ma queste coalizioni non si stabiliscono generalmente in un modo unico e determinato, risolvendosi normalmente in un susseguirsi di tradimenti, doppi giochi, inganni, e ciò non è altro che un'immagine troppo fedele del mondo dei grandi affari e di quelli strettamente connessi della politica, della diplomazia, della guerra. A lungo andare, anche l'affarista più brillante e spregiudicato deve aspettarsi la rovina; e se gli affaristi si stancano di tutto questo e si mettono d'accordo per vivere in pace fra loro, i grandi guadagni saranno riservati a quello fra loro che aspetterà vigilante il momento opportuno per rompere l'accordo e tradire i suoi compagni. Non vi è omeostasi di sorta. Abbiamo a che fare con i cicli affaristici di espansione e di crisi, con il succedersi di dittature e rivoluzioni, con le

Un simile governo - che Wiener annovera tanto tra le dittature fasciste, assimilabili alla “società delle formiche” [Wiener, 1950, pp. 75-76, trad. it.], quanto al comunismo sovietico e agli eccessi della società capitalistica [Wiener, 1950, p. 142, trad. it.; Heims, 1980, p. 311]⁶⁶⁹ - realizza una forma di cattiva omeostasi: soffocando le capacità comunicative degli individui, accresce infatti il grado di entropia dell'intero sistema sociale⁶⁷⁰.

Wiener riconobbe l'esistenza di una relazione profonda tra l'attività di elaborazione delle informazioni, propria dell'essere umano, e lo scopo stesso della vita. Quest'ultimo, a suo avviso, consisteva interamente nello svilupparsi come persona, cioè nel realizzare pienamente il potenziale umano di varietà e possibilità di scelte e azioni. Per conseguire tale scopo, l'individuo deve impegnarsi in diverse modalità di elaborazione delle informazioni, quali percepire, organizzare, ricordare, inferire, decidere, pianificare, agire, e così via. Lo sviluppo umano dipende dunque direttamente dall'attività di processare informazioni [Ward Bynum, 2009, p.10].

Qual è la forma di società che, secondo Wiener, permetteva al meglio lo sviluppo e la valorizzazione dell'essere umano visto come agente informazionale? Quelle in cui il *controllo* e la *comunicazione* non soffocassero le potenzialità e le specificità umane – un rischio immanente

guerre che tutti perdono, caratteristiche così salienti del nostro tempo” [Wiener, 1948, p.209 trad. it.]. Una sintetica esposizione di Wiener sul concetto di omeostasi applicabile all'individuo e alla società è in Wiener, 1951.

⁶⁶⁹ La conformazione che il capitalismo statunitense stava assumendo attingeva essa stessa alla retorica cibernetica, ed emblematici della denuncia di Wiener dei suoi eccessi sono alcuni passaggi di *The Human Use of Human Beings*. Vi si legge, ad esempio: “In questo, la nostra concezione della società differisce dalla società ideale prospettata dai fascisti e da molti magnati del mondo degli affari e della politica. Essi preferiscono una organizzazione in cui tutti i comandi provengano dall'alto senza che sia possibile nessuna reversibilità. Sotto di essi gli uomini sono stati ridotti al livello di esecutori degli ordini di un centro nervoso che pretende di essere superiore” [Wiener, 1950, p.30, trad. it.]; oppure: “Dobbiamo coltivare la fertilità del pensiero come abbiamo coltivato l'efficienza dell'amministrazione. Dobbiamo trovare un meccanismo grazie al quale un'invenzione d'interesse pubblico possa essere effettivamente sfruttata nell'interesse pubblico. Non possiamo permetterci il lusso di esaurire i cervelli della nazione come abbiamo esaurito le risorse del suolo. Non dobbiamo ridurci alla condizione di schiavi, iscritti come proprietà nei registri dei nostri imprenditori. Dobbiamo creare un sistema in cui l'adattabilità e la variabilità siano in funzione del miglioramento. Abbiamo bisogno di una organizzazione che sia attenta alle nuove invenzioni e alla nostra sempre crescente necessità di nuovi ritrovati. Se l'uomo deve continuare ad esistere, non deve più essere considerato meno importante degli affari” [Wiener, 1950, p.73, trad. it.]. Per converso, su Parsons gravò sempre un'ombra - mai del tutto chiarita - di “maccartismo” [cfr. Wrong, 1996; Heims, 1991, pp. 210-211, trad. it.].

⁶⁷⁰ L'impegno etico nella vicenda biografica di Wiener in opposizione all'uso bellico delle tecnologie che aveva contribuito a ideare e a sviluppare, e che il cibernetico del MIT ha espresso in opere come *The Human Use of Human Beings* e *God & Golem, Inc.: A Comment on Certain Points Where Cybernetics Impinges on Religion* [Wiener, 1964] è narrato in Conway, Siegelman, 2005, pp.401-451 trad. it. e dallo stesso Wiener nella sua autobiografia in Wiener, 1956, p.293-313. Una analisi puntuale e accessibile del pensiero etico-politico del matematico del MIT è in Numerico et. al, 2010, pp.33-37.

la visione di ogni entità come pattern di informazioni⁶⁷¹ – e consentissero a queste di esprimersi al meglio?

La risposta di Wiener, e questo approfondisce ulteriormente i motivi dell'interesse dei *New Communalists* per la cibernetica [Turner, 2006, p36]⁶⁷², erano le piccole comunità in cui le persone hanno rapporti informazionali e comunicativi diretti e non centralizzati tra loro. Scrive Wiener in *Cybernetics*:

Le comunità piccole e molto unite possiedono quindi un considerevole grado di omeostasi; e questo vale sia per comunità a un buon livello culturale in un paese civile sia per villaggi di tribù primitive. Per strani o addirittura repugnanti che ci possano sembrare i costumi di molti selvaggi, essi hanno generalmente un ben definito valore omeostatico, la cui interpretazione costituisce parte della funzione degli antropologi. È solo nella grande comunità, dove i Signori delle Cose come Stanno si proteggono dalla fame con la ricchezza, dalla pubblica opinione con la privacy e l'anonimato, dalla critica privata con le leggi contro la diffamazione e con il possesso dei mezzi di comunicazione, che la spietatezza può raggiungere i suoi più alti livelli. Fra tutti questi fattori antiomeostatici della società, il controllo dei mezzi di comunicazione è il più efficace e importante [Wiener, 1948, pp.210-211 trad. it.].

La metafora cibernetica dell'analogia funzionale con il cervello trovava il proprio culmine tecnologico e implementativo, lungo il percorso fin qui ricostruito, prima nel circuito integrato e poi nel microprocessore. Da un lato, essa completava la riduzione delle facoltà logiche a operazioni formalizzabili e replicabili nei dispositivi elettronici; dall'altro offriva, in modo "paradossale", strumenti di ampliamento del potenziale informazionale di ciascun individuo, concretizzando sul piano socio-tecnico le promesse di "liberazione" immaginate dagli esponenti della controcultura. L'elemento umano, inizialmente affiancato al calcolatore e successivamente reso quasi indistinguibile da esso nei sistemi integrati, trovava un

⁶⁷¹ "Al pari di ogni altro ente fisico", scrive Ward Bynum, per Wiener "gli esseri umani possono essere visti come enti fondamentalmente informazionali. Come gli altri animali, sono essenzialmente *insiemi* di informazioni di Shannon che persistono attraverso un continuo scambio di materia-energia" [Ward Bynum, 2009, p.8].

⁶⁷² "Come Norbert Wiener due decenni prima, molti nella controcultura videro nella cibernetica una visione di un mondo costruito non attorno a gerarchie verticali e a flussi di potere dall'alto verso il basso, ma attorno a circuiti ricorsivi di energia e informazione. Questi circuiti offrivano la possibilità di un ordine sociale stabile fondato non sulle catene di comando psicologicamente opprimenti che caratterizzavano la vita militare e aziendale, ma sul flusso e riflusso della comunicazione" ["Like Norbert Wiener two decades earlier, many in the counterculture saw in cybernetics a vision of a world built not around vertical hierarchies and top-down flows of power, but around looping circuits of energy and information. These circuits presented the possibility of a stable social order based not on the psychologically distressing chains of command that characterized military and corporate life, but on the ebb and flow of communication, Turner, 2006, p.36 trad. mia]

rispecchiamento nel design dell'architettura di von Neumann, strutturata in connessione seriale con i processi informazionali dell'operatore.

Con la microelettronica si apriva invece un nuovo orizzonte: la personalizzazione della computazione permetteva un accesso sempre più individuale e diffuso all'universo digitale - già inaugurato dai primi calcolatori e da esperimenti come le simulazioni Monte Carlo - e alimentava modalità di comunicazione e di elaborazione informazionale via via più pervasive, diversificate e capillari.

È possibile adesso andare ad indagare - servendoci anche degli strumenti di strumenti concettuali e di analisi che hanno molto in comune con la prospettiva informazionale e cibernetica [Cfr Bynum, 2009; Bynum 2010]⁶⁷³ - il mondo "digitale" che si è dischiuso con Monte Carlo e che, negli sviluppi culturali e tecnologici successivi, i computer hanno contribuito a personalizzare nell'accesso. Metteremo in luce, da un lato, il materiale di cui esso è composto, ampliando e approfondendo i concetti di dato, definendo i *Big Data*, e informazione finora proposti; dall'altro, i modi della relazione dell'essere umano con questo nuovo universo, in coerenza con la prospettiva storica e scientifica sin qui delineata.

5. Un universo di informazioni: dati e interfacce

Il processo innescato dalle simulazioni Monte Carlo, che ha trasformato il calcolatore "da strumento a natura", ha reso possibile che "attraverso la tecnologia digitale possiamo ricreare o, in modo ancora più radicale, generare la realtà con cui interagiamo, prima codificandola nel linguaggio dei bit e poi traducendo ulteriormente il risultato digitale in entità analogiche facilmente gestibili dalla nostra mente" [Floridi, 1999, p. 8, trad. mia]⁶⁷⁴. Se il calcolatore è ciò che permette di studiare il mondo, perché implementa metodi che ne rispecchiano la natura profonda, ed è al contempo capace di generarne uno artificiale altrettanto significativo,

⁶⁷³ La prospettiva filosofica in questione è quella della Filosofia dell'Informazione proposta da Luciano Floridi e dei suoi sviluppi. [Floridi, 1999; 2009; 2011], che Bynum mette a confronto in modo serrato con il pensiero cibernetico di Wiener, evidenziandone analogie e differenze di fondo. Punto di partenza comune è la concezione secondo cui "l'universo è essenzialmente informazionale - costituito di oggetti e processi informazionali che interagiscono in modo dinamico. Entrambi considerano gli esseri umani come oggetti informazionali e affermano che l'entropia rappresenti un male significativo nel mondo" [Bynum, 2009, p. 22]. Ciò non deve però far dimenticare le divergenze tra il "materialista" Wiener e il "platonico" Floridi [cfr. Bynum, 2009, pp. 22-23].

⁶⁷⁴ "By means of digital technology we can recreate or, even more radically, generate the reality we deal with, first by codifying it into the language of bits and then by further translating the digital outcome into analogue entities easily manageable by our minds".

offrendo modalità di accesso personali e progressivamente ubiquie [*ubiquitous*] [Weiser, 1991]⁶⁷⁵, è necessario prendere in esame le relazioni tra queste due dimensioni.

La descrizione informatizzata e il controllo dell'ambiente fisico, insieme alla costruzione digitale di un mondo sintetico, sono infine intrecciati con una quarta area di applicazione, rappresentata dalla trasformazione del macrocosmo enciclopedico di dati, informazioni, idee, conoscenze, credenze, esperienze codificate, memorie, immagini, interpretazioni artistiche e altre creazioni mentali in una *infosfera* globale [Floridi, 199, p.8 trad. mia enfasi mia]⁶⁷⁶.

Per comprendere meglio il concetto di *infosfera* bisogna richiamare quello di *Livello di Astrazione* (LoA/LdA) incontrato in precedenza e cioè, un insieme non vuoto di variabili tipizzate interpretate (osservabili) che costituiscono l'accesso, in relazione al nostro scopo epistemico, ad un oggetto (il sistema) per la costruzione di un modello del sistema in questione [Floridi, 2008]. Con *variabile* indichiamo, come è d'uso comune nella scienza, "un simbolo che sta in luogo di un referente" [Floridi, 2019, p.58 trad. it.]. La variabile *tipizzata* è quella che assume i suoi valori solo da uno specifico insieme, "che può contenere soltanto una determinata tipologia di dati". Ad esempio, nel chiedere a qualcuno il numero di telefono, ci si attende che il *tipo* [*type*] della risposta fornita sia quello dei numeri naturali [Floridi, 2008; Floridi, 2019, p.58 trad. it.]. Un *osservabile*, invece, "non è necessariamente inteso come derivato da misurazioni quantitative o percezione empirica" [Floridi, 2008 trad. mia]⁶⁷⁷, ma è una è una variabile

⁶⁷⁵ Il termine *Ubiquitous computing* è stato coniato e reso celebre dal capo del Computer Science Laboratory dello Xerox Palo Alto Research Center, Mark Weiser [1952–1999]. Con i computer ubiqui Weiser intendeva quelle tecnologie "che scompaiono. Esse si intrecciano nel tessuto della vita quotidiana fino a diventare indistinguibili da essa" ["the most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it", Weiser, 1991 trad. mia], proponendo un ulteriore superamento rispetto alla personalizzazione - da lui considerata soltanto come la seconda ondata della tecnologia computazionale, successiva alla prima, quella dei mainframe. "Tali macchine [i PC] non possono davvero rendere l'informatica una parte integrante e invisibile della vita delle persone. Stiamo quindi cercando di concepire un nuovo modo di pensare i computer, che tenga conto del mondo umano e consenta ai computer stessi di svanire sullo sfondo" ["Such machines cannot truly make computing an integral, invisible part of people's lives. We are therefore trying to conceive a new way of thinking about computers, one that takes into account the human world and allows the computers themselves to vanish into the background", Weiser, 111, trad. mia]. Gli sviluppi del Web, dell'Internet of Things, la pervasività del digitale e la miniaturizzazione delle interfacce sembrano confermare le intuizioni, allora visionarie, di Weiser. Sulla sua figura si veda Tinnell, 2023; sugli sviluppi pionieristici dello Xerox PARC, che tuttora plasmano la forma delle tecnologie digitali e dell'informazione, si vedano Haigh, Ceruzzi, 2021, pp. 244–253; Bardini, 2000, pp. 142–180, 215–228; Rheingold, 2000.

⁶⁷⁶ "The computerised description and control of the physical environment, together with the digital construction of a synthetic world, are, finally, intertwined with a fourth area of application, represented by the transformation of the encyclopaedic macrocosm of data, information, ideas, knowledge, beliefs, codified experiences, memories, images, artistic interpretations and other mental creations into a global infosphere".

⁶⁷⁷ "Being an abstraction, an observable is not necessarily meant to result from quantitative measurement or even empirical perception".

tipizzata *interpretata*, “vale a dire una variabile tipizzata unitamente all’affermazione di quale sia la caratteristica del sistema in esame che essa rappresenta”. Per rimanere all’esempio precedente, “un insieme di dati può avere i *numeri naturali* come tipo e il *numero telefonico* come caratteristica del sistema” [Floridi, 2019, p.58 trad.it.].

Il Livello di Astrazione, inteso come insieme di osservabili di un sistema – vale a dire il referente e la fonte delle informazioni dell’analisi – è ciò che “qualifica il livello al quale un sistema è considerato” [Floridi, 2019, p. 59, trad. it.].

È possibile spiegare questo metodo d’analisi, derivato da quel “settore dell’informatica, chiamato *metodi formali*, nel quale la matematica discreta è usata per specificare e analizzare il comportamento di sistemi informativi” [Floridi, 2019, p.56 trad. it.]⁶⁷⁸, attraverso un semplice esempio fornito dallo stesso Floridi:

Supponiamo di unirvi a Alice, Bob e Carola alla festa. Si trovano nel mezzo di una conversazione. Alice è una collezionista e una potenziale acquirente, Bob ama riparare le cose nel suo tempo libero e Carola è un’economista. Non conosciamo il tema della conversazione, ma questo è quanto siamo in grado di sentire: – Alice osserva che qualcosa (qualsiasi sia questo “qualcosa”) ha un dispositivo antifurto installato, è parcheggiato in garage quando non è utilizzato ed è di proprietà di un solo soggetto; – Bob osserva che il suo motore non è quello originale, che la sua carrozzeria è stata da poco riverniciata, ma che le parti in pelle sono molto consumate; – Carola osserva che il vecchio motore consumava troppo, che ha un valore stabile di mercato e che le sue parti di ricambio sono costose [...] I partecipanti alla discussione guardano al sistema in esame – il “qualcosa” – a partire dai loro interessi, che orientano la scelta delle loro interfacce concettuali o, più precisamente, dei loro livelli di astrazione. Nell’esempio visto, il LdA di Alice può consistere di osservabili relativi alla sicurezza, al metodo di conservazione e

⁶⁷⁸ Sull’impatto che la tecnologia computazionale, l’informatica e il digitale ha avuto nel “rinnovamento” della filosofia in termini di metodologie e oggetti di indagine si veda, (ed.) Moor, Bynum, 1998. Come scrivono i due filosofi nell’introduzione all’opera: “L’informatica fornisce alla filosofia un insieme di nozioni semplici, ma incredibilmente feconde – nuovi e in evoluzione *soggetti di indagine, metodi e modelli per l’indagine filosofica*. L’informatica porta nuove opportunità e sfide alle attività filosofiche tradizionali. Di conseguenza, essa sta cambiando le attività professionali dei filosofi, incluso il modo in cui fanno ricerca, come cooperano tra loro e come insegnano i loro corsi. Soprattutto, l’informatica sta cambiando il modo in cui i filosofi comprendono concetti fondamentali della filosofia, come mente, coscienza, esperienza, ragionamento, conoscenza, verità, etica e creatività. Questa tendenza nell’indagine filosofica che incorpora l’informatica come oggetto, metodo o modello ha acquisito un slancio costante. *Una Fenice Digitale sta sorgendo!*” [“Computing provides philosophy with such a set of simple, but incredibly fertile, notions - new and evolving subject matters, methods, and models for philosophical inquiry. Computing brings new opportunities and challenges to traditional philosophical activities. As a result, computing is changing the professional activities of philosophers, including how they do research, how they cooperate with each other, and how they teach their courses. Most importantly, computing is changing the way philosophers understand foundational concepts in philosophy, such as mind, consciousness, experience, reasoning, knowledge, truth, ethics and creativity. This trend in philosophical inquiry that incorporates computing in terms of a subject matter, a method, or a model has been gaining momentum steadily. A Digital Phoenix is rising!”, Bynum, Moor, 1998, p.1 trad. mia enfasi mia].

alla storia del proprietario; quello di Bob, di osservabili relativi alle condizioni del motore, alle condizioni della carrozzeria e alle condizioni interne; infine, quello di Carola, di osservabili relativi ai costi di gestione, al valore di mercato e ai costi di manutenzione [Floridi, 2019, pp.58-59 trad.it.].

Un LoA può assumere diversi gradi di astrazione a seconda del numero di osservabili che prende in considerazione, i quali possono risultare correlati, distinti o intrecciati, senza che vi sia necessariamente un ordine gerarchico. Si parla di livello più concreto, o più basso, quando l'analisi include un maggior numero di osservabili; al contrario, un livello più astratto, o più elevato, corrisponde a una descrizione meno granulare, che trascura alcuni dettagli disponibili nei livelli inferiori.

Ad un determinato Livello di Astrazione, "ogni processo, operazione, cambiamento, azione ed evento può essere trattato come oggetto informativo". L'informazione, ciò che i calcolatori (e i cervelli) processano, è una sorta di *livello ontologico minimale* che caratterizza ogni ente: "Un ente è un pacchetto coerente di informazioni, il quale non contiene in sé contraddizione e può ricevere un nome o una denotazione in un processo informativo" [Floridi, 2009, p.39]. Da questa prospettiva, "secondo Floridi, ogni ente esistente nell'universo, se considerato a un certo 'livello d'astrazione', è un 'oggetto informazionale' con una caratteristica struttura di dati che ne costituisce la sua vera natura" [Ward Bynum, 2009, p.17].

L'*infosfera*⁶⁷⁹ - che nasce dall'incrocio tra la capacità del calcolatore di modellizzare gli oggetti come oggetti informazionali e quella di creare uno spazio digitale sintetico - è dunque "l'ambiente costituito dalla totalità degli enti informazionali, che includono tutti gli agenti, i processi, le loro proprietà e relazioni reciproche" [Floridi, 2009, p.37].

Ma questo è solo un "*livello minimo*". "A un *livello massimo*, l'infosfera è un concetto che può essere utilizzato anche come sinonimo di realtà, laddove interpretiamo quest'ultima in termini informazionali. In tal caso, l'idea è che ciò che è reale è informazionale e ciò che è informazionale è reale" [Floridi, 2014, pp.44-45 trad. it.]. Qui si palesa in modo evidente il

⁶⁷⁹ Il primo impiego del termine sembra essere quello dell'economista Kenneth E. Boulding (1910–1993), che nel 1970 scrive: "L'*infosfera* consiste quindi negli input e negli output delle conversazioni, dei libri, della televisione, della radio, dei discorsi, delle funzioni religiose, delle lezioni e delle conferenze, così come delle informazioni ricevute dal mondo fisico tramite l'osservazione personale" [The infosphere then consists of inputs and outputs of conversation, books, television, radio, speeches, church services, classes, and lectures as well as information received from the physical world by personal observation, Boulding, 1970, p. 16, trad. mia]. L'*infosfera*, secondo Boulding, si affianca alle altre *sfe* che compongono il sistema mondo nella sua globalità: la *litosfera* studiata dai geologi, l'*idrosfera* dagli oceanografi, l'*atmosfera* dai meteorologi, la *biosfera* dai biologi e la *sociosfera* dagli scienziati sociali [cfr. Boulding, 1970, p. 1].

passaggio del computer da strumento a natura e la metamorfosi del programmatore come *designer* della realtà e dei modi di relazionarsi a questa.

È possibile ora indagare brevemente sul materiale che costituisce questa realtà, i *dati*. Abbiamo precedentemente, trattando le vicende riguardanti la nascita di ENIAC, fornito una definizione *diaforica* dei dati intesi come una differenza, come una mancanza di uniformità tra due variabili, che ora siamo in grado di completare. Questi possono essere divisi in: - *dati primari*: i dati principali che un sistema raccoglie e trasmette per fornire informazioni dirette all'utente. "Normalmente, quando si parla di dati, e delle corrispondenti informazioni che costituiscono, si fa implicitamente riferimento a dati e informazioni primari". Un esempio è la luce rossa che segnala la batteria scarica: essa rappresenta un dato primario, da cui si ricava un'informazione primaria, cioè la necessità di ricarica; - *dati secondari*: "Sono il contrario dei dati primari, costituiti dalla loro assenza", ad esempio, il silenzio di un cane nel racconto di Sherlock Holmes è interpretato come un dato secondario, capace di produrre informazione (in quel caso, sull'identità del colpevole); - *metadati*: "Sono indicazioni sulla natura di altri dati (di solito primari). Descrivono proprietà come la posizione, il formato, l'aggiornamento, la disponibilità, le restrizioni d'uso ecc.". Servono a contestualizzare i dati primari e a renderli interpretabili; - *dati operativi*: "Sono i dati che riguardano le operazioni dell'intero sistema di dati e del suo funzionamento". Rimanendo all'esempio della batteria, una spia gialla che segnala un errore nel sistema di controllo avverte che anche l'indicatore della batteria potrebbe essere malfunzionante, modificando così l'interpretazione di altri dati.; - *dati derivati*: "Si tratta di dati che possono essere estratti da altri dati quando questi ultimi sono utilizzati come fonti indirette alla ricerca di schemi, indizi o inferenze relativamente ad aspetti diversi da quelli affrontati direttamente dai dati stessi, per esempio per compiere analisi comparative e quantitative (ideometria)". Un esempio è la localizzazione di una persona a partire dalla ricevuta della carta di credito: il dato non era stato registrato con quell'intento, ma può essere usato per ricavare informazioni secondarie, spesso a fini investigativi o commerciali. [Floridi, 2011, pp.87-88, trad. it. Floridi, 2024].

Questa classificazione qualitativa si può allargare ad una quantitativa che sottolinea meglio l'ampiezza e la portata dell'universo dischiusosi con le simulazioni digitali. Come detto in precedenza, dai 700 bit della memoria iniziale dell'ENIAC, l'universo digitale si è espanso in maniera esponenziale. Il 2003, gli esseri umani avevano creato 5 exabyte (10^{18}) di dati. Nel 2012 la stessa quantità era prodotta ogni due giorni, sino a raggiungere i 2,72 zettabyte dello

stesso anno [Sagiroglu, Sinanc, 2013], cresciuti a 7 il 2014 [Hariri et al, 2019]. Oggi, il 2025, si sfiorano i 200 ZB di dati.

Questa enorme massa di dati, che lo sviluppo delle tecnologie digitali, in termini di *mining* e produzione, incrementa in modo vertiginoso, sono i cosiddetti *Big Data*. Per definire questi ultimi⁶⁸⁰ si è ricorso, da parte di analisti e studiosi, al concetto della lettera “V”, l’iniziale che ne indica le caratteristiche [Fosso Wamba et al., 2015]. Una iniziale definizione dei Big Data secondo questo modello si può far risalire all’analista statunitense Douglas Laney, che nel 2001 introdusse, analizzando le sfide che la crescente mole di dati lanciava in ambito aziendale, commerciale e informatico, il modello delle 3V: *Volume*, *Velocity* e *Variety* [Laney, 2001].

Secondo questa prima classificazione, il *Volume* indica la grande quantità di dati che richiede enormi capacità di archiviazione o che consiste in un numero molto elevato di record; la *Velocity* si riferisce alla frequenza o alla rapidità con cui i dati vengono generati e/o trasmessi [Fosso Wamba et al., 2015; Russom, 2011] la *Variety*, infine:

È ciò che rende i Big Data davvero “big”. I Big Data provengono da un’enorme varietà di fonti e si presentano generalmente in tre forme: strutturati, semi-strutturati e non strutturati. I dati strutturati vengono inseriti in un data warehouse già etichettati e facilmente ordinabili, mentre i dati non strutturati sono casuali e difficili da analizzare. I dati semi-strutturati, invece, non rispettano campi fissi, ma contengono tag che separano gli elementi informativi [Sagiroglu, Sinanc, 2013 trad. mia]⁶⁸¹.

Accanto a queste prime tre “V”, volume, velocità e varietà, se ne sono sommate via via altre. Man mano che il valore economico dei dati è cresciuto, si è aggiunta una quarta V, quella del *Valore (Value)*, che indica “il grado in cui i Big Data generano intuizioni economicamente significative e/o benefici attraverso l’estrazione e la trasformazione” [Fosso Wamba et. al, 2015

⁶⁸⁰ Una panoramica delle definizioni è in Fosso Wamba et al., 2015 Table.1 p.236. Intorno all’origine del nome si rimanda a Diebold, 2012. Il termine “Big Data”, nato all’incrocio tra informatica, statistica ed econometria, sembra avere origine un’origine informale, nelle conversazioni alla mensa della *Silicon Graphics Inc.* (SGI) a metà degli anni ’90, dove fu centrale la figura dell’informatico e dirigente tecnologico, John Mashey. I primi riferimenti accademici rilevanti appaiono nel 1998, con una pubblicazione dell’informatico e statistico Sholom Weiss e dell’esperto di Machine Learning Nitin Indurkha nell’ambito dell’informatica [Weiss, Indurkha, 1998]; e nel 2000 con un contributo in ambito statistico-econometrico dello stesso Francis X. Diebold. Nel 2001, una nota tecnica non pubblicata di Douglas Laney per la Gartner, contribuì e diffondere il concetto. L’espressione “Big Data” può quindi, secondo Dan essere ragionevolmente attribuita a Mashey, Weiss, Indurkha, Laney e allo stesso Diebold.

⁶⁸¹ “Variety makes big data really big. Big data comes from a great variety of sources and generally has in three types: structured, semi structured and unstructured. Structured data inserts a data warehouse already tagged and easily sorted but unstructured data is random and difficult to analyze. Semistructured data does not conform to fixed fields but contains tags to separate data elements”.

p. 236 Tab.1 trad. mia]⁶⁸². A questa si è affiancata una quinta dimensione, la *Veracità* (*Veracity*), proposta per evidenziare l'importanza della qualità dei dati e del livello di fiducia nelle diverse fonti da cui essi provengono [White, 2012].

La crescita della classificazione non si è fermata. Nel 2014, l'astrofisico e data analyst Kirk Borne ha proposto una classificazione a 10 V per i Big Data, a cui, alle 5 già incontrate, ne ha aggiunto ulteriori 5 da interpretare come "sfide" che i dati lanciano, e cioè la *Variabilità* (*Variability*), ovvero l'instabilità dei dati nel tempo e nei contesti; la *Visualizzazione* (*Visualization*), ossia la difficoltà di rendere i dati comprensibili attraverso rappresentazioni grafiche efficaci; la *Volatilità* (*Volatility*), cioè la breve durata dell'utilità di certi dati prima che diventino obsoleti; la *Validità* (*Validity*), che riguarda la coerenza e correttezza dei dati rispetto allo scopo per cui vengono usati; e infine la *Vulnerabilità* (*Vulnerability*), che indica i rischi legati alla sicurezza, alla privacy e alla protezione delle informazioni sensibili [Borne, 2014].

Queste ulteriori V, intese come fattori critici da considerare nella gestione dei dati, riflettono la crescente attenzione verso le dimensioni dell'affidabilità epistemica, così come verso le implicazioni etiche e legali connesse all'uso dei *Big Data*.

Il numero delle caratteristiche non si è ancora arrestato. In tempi recenti sono state proposte classificazioni che includono 14, o persino 17 V [Panimalar et al., 2017], un numero destinato con ogni probabilità ad aumentare nel tempo.

Ma questa enorme mole di dati è informazione? Ritorna qui l'importanza dei Livelli di Astrazione.

I dati non sono mai accessibili ed elaborati (da un agente informazionale) indipendentemente da un livello di astrazione. Intesi come enti relazionali, i dati sono *possibilità vincolanti*: permettono o invitano alcuni costrutti (in tal senso, sono possibilità per l'agente informazionale che può sfruttarle) e ne contrastano o impediscono altri (sono vincoli per lo stesso agente), in ragione della tipologia di interazione e della natura dell'agente informazionale che li elabora [Floridi, 2011, p.87 trad. it. Floridi, 2025].

Caratterizzati come possibilità vincolanti, i dati possono essere impiegati come input per formulare adeguatamente delle domande solo se vengono collocati entro un determinato Livello di Astrazione (LoA). È infatti il LoA a orientare l'input verso un significato specifico,

⁶⁸² "The extent to which big data generates economically worthy insights and/or benefits through extraction and transformation".

trasformandolo così in informazione come output. Non deve stupire allora che l'informazione dotata di significato⁶⁸³ possa essere definita "*dati + domande*" [Floridi, 2011, p.87 trad.it Floridi, 2024 enfasi mia]. Queste domande sono espresse ad un livello di astrazione. Sono questi ultimi, e l'accesso che ci consentono ai dati, a renderli informativi e dotati di senso.

Un insieme di Livelli di Astrazione costituisce un'"*interfaccia*" [Floridi, 2019, p. 58, trad. it. enfasi mia]. L'interfaccia tra utente e dati è il medium che consente di gestire e ricavare informazione da questi ultimi. In questa prospettiva si comprende come l'espansione dell'universo della simulazione - di cui costituisce esempio quanto testimoniato da Metropolis a proposito del metodo Monte Carlo - sia stata accompagnata e resa possibile da continui miglioramenti nelle modalità di relazione con esso, tali da renderlo comprensibile e manipolabile dall'essere umano. L'elaborazione computazionale di dati complessi, infatti, non avrebbe senso se questi non fossero accessibili a qualche livello a un essere umano in grado di interpretarli.

L'interfaccia di un computer rappresenta un esempio privilegiato di accesso a un sistema attraverso un insieme di Livelli di Astrazione:

Il concetto di interfaccia in un sistema informatico può essere utile per illustrare cosa sia un LoA (Livello di Astrazione). Un'interfaccia può essere descritta come un elemento intra-sistemico che trasforma gli output del sistema A (ad esempio un computer) negli input del sistema B (ad esempio un utente umano), e viceversa, producendo un cambiamento nei tipi di dati. I livelli di astrazione sono paragonabili alle interfacce per due motivi: sono concettualmente collocati tra le fonti di dati e gli spazi

⁶⁸³ Alla base di questa concezione vi è una teoria dell'informazione semantica in senso forte, che deve includere nella sua definizione le veridicità. I dati che la costituiscono, secondo questa teoria, come possibilità vincolanti non possono essere solo dotati di senso e ben formati. Devono essere anche veritieri. Considerati:

- SDI (*standard definition of information*);
- DOS (*declarative, objective, semantic*);
- σ (infon)= simbolo usato per riferirsi a elementi discreti di informazione, a prescindere dal loro codice semiotico e dalla loro implementazione fisica;
- δ = dati aleticamente non qualificati, ben formati e dotati di significato:

L'informazione semantica in senso forte (RSDI) sarà così definita:

RSDI) σ è un'istanza di informazione DOS se e solo se:

1. σ consiste di n dati (d), per $n \geq 1$;
2. i dati sono ben formati (wfd);
3. i wfd sono significativi ($mwfd = \delta$);
4. i δ sono veritieri. [Floridi, 2011, pp.92-107]. La teoria secondo cui l'informazione implicherebbe necessariamente la veridicità è stata contestata da Scarantino e Piccinini, 2010, i quali ne mettono in evidenza la circolarità e cercano di dimostrare che le nozioni di informazione impiegate nelle scienze cognitive e nell'informatica consentono che un soggetto A possa avere informazioni circa l'avvenire di p anche quando p è falso. Per quanto riguarda il dibattito sul concetto di informazione in Floridi, si veda ed. Allo, 2010. Ulteriori critiche a tale teoria sono state mosse da Ferguson, 2016, che propone due paradossi per confutarla, e da Lundgren, 2017. Per una panoramica più ampia sulla filosofia dell'informazione si rimanda a Adriaans, 2020.

informativi degli agenti; e rappresentano il punto in cui sistemi (diversi) e indipendenti si incontrano, interagiscono o comunicano tra loro [Floridi, 2011, p.52 trad. mia]⁶⁸⁴.

Jonathan Grudin (1949–vivente), studioso di *Human-Computer Interaction* (HCI) e figura di riferimento presso Microsoft per la ricerca in questo ambito, fa risalire l'inizio della storia del rapporto tra essere umano e informazione ai sistemi pre-digitali sviluppati tra la fine dell'Ottocento e l'inizio del Novecento per la gestione libraria. In quegli anni, grazie alla riduzione dei costi di carta, stampa e trasporto, aumentò la diffusione dell'informazione veicolata dai libri e, parallelamente, si consolidarono la definizione e lo statuto scientifico delle professioni legate al mondo librario.

Negli Stati Uniti nacquero migliaia di biblioteche pubbliche, spesso poco finanziate, che adottarono il sistema di classificazione decimale Dewey e quello di classificazione della Libreria del Congresso⁶⁸⁵. Nel 1926 la *Carnegie Foundation* istituì la *Graduate Library School* presso l'Università di Chicago, l'unica che conferiva dottorati in studi bibliotecari. Negli USA prevalse un approccio volto alla gestione pragmatica - non a caso posta in relazione da Grudin con l'organizzazione tayloristica della fabbrica e agli studi di psicologia del lavoro di Lilian Gilbreth (1878-1972) - delle biblioteche e alla formazione di specialisti in biblioteconomia. [Cfr. Grudin 2017, pp.7-9].

Questa gestione volta principalmente ai bisogni degli specialisti [Burke, 2007], era spesso accompagnata da forme di meccanizzazione, di cui sono un esempio le schede perforate Hollerith che prese in esame in precedenza.

⁶⁸⁴ "The concept of interface in a computer system may be helpful in illustrating what an LoA is. An interface may be described as an intra-system, which transforms the outputs of system A (e.g. a computer) into the inputs of system B (e.g. a human user) and vice versa, producing a change in data types. LoAs are comparable to interfaces for two reasons: they are conceptually positioned between data sources and the agents's information spaces; and they are the place where (diverse) independent systems meet, act upon or communicate with each other".

⁶⁸⁵ Il *Dewey Decimal Classification* (DDC), il sistema decimale Dewey fu sviluppato da Melvil Dewey (1851-1931), bibliotecario e riformatore statunitense, nel 1876 come sistema per organizzare i libri nelle biblioteche secondo dieci grandi classi tematiche numerate da 000 a 999. Ogni classe era suddivisa gerarchicamente in sottosezioni decimali sempre più specifiche. Il DDC introdusse due innovazioni fondamentali, rispetto ai sistemi precedenti basati sull'acquisizione del volume: la collocazione relativa e l'indice che ne permettevano la collocazione in base al contenuto. La notazione si basa su tre cifre per le classi principali, con l'uso di decimali per articolare ulteriori dettagli specifici. Il metodo è esposto in Dewey, 1876. Il *Library of Congress Classification* (LCC), invece, è un sistema di organizzazione dei libri sviluppato dalla Libreria del Congresso USA alla fine dell'Ottocento e introdotto ufficialmente nel 1901. Il sistema Dewey era pensato per le biblioteche pubbliche, l'LCC, invece, fu progettato per rispondere alle esigenze accademiche. Si basa su una combinazione alfanumerica: le lettere indicano le grandi aree disciplinari mentre i numeri permettono una classificazione più precisa delle sotto-caratteristiche all'interno di ciascun ambito, rendendolo maggiormente flessibile ed adattabile a corpus di testi in crescita.

I primi tentativi originali di elaborazione e gestione meccanizzata dell'informazione si sono rivelati fallimentari, «ma non perché le schede perforate o altri dispositivi simili fossero impraticabili, bensì perché un'impresa del genere richiedeva un approccio molto più radicale rispetto alla semplice estensione meccanica». Questa “svolta radicale” consiste in due fattori, «nel linguaggio e nella fisica dell'infosfera», che si incontrano e si intrecciano nell'universo aperto con la scintilla delle implementazioni Monte Carlo: la “logica binaria”, il passaggio dall'analogico al digitale e dal continuo al discreto, e, attraverso gli sviluppi che abbiamo via via analizzato, «l'implementazione elettronica di tale logica attraverso il microprocessore» [Floridi, 1999, p. 13, trad. mia]⁶⁸⁶

Quest'ultima prese forma inizialmente nell'analogia funzionale tra i processi logici del pensiero e del cervello e i loro corrispettivi artificiali (la metafora cibernetica), attraverso la transizione dall'elettromeccanica alle valvole termoioniche e ai tubi catodici - che consentivano una prima, rudimentale, visualizzazione dei processi computazionali, come nel calcolatore di Manchester o nel display dell'ENIAC - fino agli sviluppi successivi dell'elettronica a stato solido.

Il passaggio alla “nuova fisica della conoscenza” [Floridi, 1999, p.13 trad. mia]⁶⁸⁷, dall'informazione analogica scritta o stampata⁶⁸⁸ su carta ai dati digitali ha permesso di controllare, gestire – oltre che ad espandere – il sistema informazionale della conoscenza stessa.

“Le schede perforate”, scrive Floridi, “acquisirono il loro grande valore solo in quanto dispositivi di input per una macchina binaria ed elettronica”. Su di esse era possibile rendere in forma digitale i processi più complessi che riguardavano la struttura stessa della realtà, come quelli delle reazioni particellari e dei processi stocastici implicati. Da questo momento, “le

⁶⁸⁶ “Early efforts to construct merely mechanical devices to process information efficiently all failed, but the reason why they failed is not that punched cards or other similar devices were unpractical, but that such an enterprise required a much more radical approach than the simple extension of a mechanical solution to the problem of semiotic management. A modification both in the language of the infosphere – binary logic – and in its physical nature – the electronic implementation of such a logic via the microprocessor – was necessary”.

⁶⁸⁷ “Only a new physics of knowledge, the passage from printed paper to digital-electronic data, finally made possible a thoroughly satisfactory way of managing information, and hence much more efficient control over the system of knowledge”.

⁶⁸⁸ La svolta dall'analogico della carta al bit del digitale richiama inevitabilmente il campo di studi delle *Digital Humanities*, ovvero l'insieme delle attività di ricerca che utilizzano le tecnologie digitali - compresa l'intelligenza artificiale - applicate allo studio delle discipline umanistiche. Oltre alla digitalizzazione di contenuti come testi, immagini e documenti, le *Digital Humanities* comprendono l'analisi computazionale dei testi, la ricerca e visualizzazione dell'informazione, la filologia computazionale, l'analisi automatica del linguaggio, la costruzione di archivi digitali, l'edizione elettronica critica e la modellazione di dati storici e culturali. In Italia, un'esperienza pionieristica fu quella di padre Roberto Busa (1913–2011), che negli anni Cinquanta avviò, in collaborazione con IBM, l'informatizzazione dell'*Index Thomisticus*. Oggi il settore ha un riferimento istituzionale nell'Associazione per l'Informatica Umanistica e la Cultura Digitale (AIUCD). Sul tema si rimanda a Numerico et al., 2010; Ciotti (a cura di), 2014; e Ciotti, 2023.

tecnologie dell'informazione e della comunicazione (ICT), come tanto attesa risposta all'invenzione della stampa, divennero rapidamente la più pervasiva di tutte le tecnologie del nostro tempo" [Floridi, 1999, p.13 trad.mia]⁶⁸⁹

L'espansione dell'infosfera che abbiamo quantificato e caratterizzato sopra parlando di *Big Data*, riguarda dati che "vengono generati da transazioni online, email, video, audio, immagini, flussi di clic, registri, post, query di ricerca, cartelle cliniche, interazioni sui social network, dati scientifici, sensori, telefoni cellulari e relative applicazioni". Dati che sono generati, con sviluppo esponenziale, dalle ICT. L'insieme di questi dati vanno a comporre "database che crescono in modo massiccio e diventano difficili da acquisire, strutturare, memorizzare, gestire, condividere, analizzare e visualizzare tramite gli strumenti tradizionali [...]" [Sagiroglu, Sinanc, 2013 trad.mia]⁶⁹⁰.

"La lezione da trarre da un simile processo è chiara", sostiene Floridi, "Se da un lato l'emergere e la diffusione della tecnologia informatica ha accelerato l'espansione della conoscenza - già avviata con l'invenzione della stampa - [...] dall'altro le ICT hanno anche cominciato a offrire almeno una soluzione parziale ai problemi organizzativi derivanti dall'immenso dominio della conoscenza codificata" [Floridi, 1999, p.13-14 trad. mia]⁶⁹¹.

I modi di interfacciarsi e interagire con l'immensa massa di dati, affinché questi diventino informazioni significative, è uno dei modi con cui è avvenuta la ristrutturazione della conoscenza. "Il processo necessario di conversione dell'intero dominio della conoscenza organizzata in un nuovo macrocosmo digitale cominciò lentamente negli anni '50" [Floridi, 1999, p. 14 trad. mia]⁶⁹². Questa affermazione va letta insieme a quella dello storico del *computing* Paul Ceruzzi, secondo il quale "gli anni Quaranta furono un decennio in cui vennero

⁶⁸⁹ "Punched cards acquired their great value only as input devices of a binary and electronic machine. Once this step was made, ICT, as the long-awaited response to the invention of printing, quickly became the most pervasive of all the technologies of our time".

⁶⁹⁰ "These data are generated from online transactions, emails, videos, audios, images, click streams, logs, posts, search queries, health records, social networking interactions, science data, sensors and mobile phones and their applications They are stored in databases grow massively and become difficult to capture, form, store, manage, share, analyze and visualize via typical [...] tools".

⁶⁹¹ "The lesson to be learnt from such a process is clear. If on the one hand the appearance and spread of computer technology has caused the expansion of knowledge, already begun with the invention of printing, to accelerate [...] on the other hand ICT has also begun to provide at least a partial solution to the organisational problems arising from the immense domain of codified knowledge".

⁶⁹² "The necessary process of converting the entire domain of organised knowledge into a new, digital macrocosm began slowly in the 1950s".

sollevati interrogativi fondamentali sul ruolo adeguato dell'interazione tra esseri umani e macchinari di controllo complessi" [Ceruzzi, 2012, p.42 trad. mia]⁶⁹³.

È in quest'ottica che si colloca la scelta metodologica di riservare un'attenzione specifica allo sviluppo storico e tecnologico dei calcolatori, dopo aver analizzato la teoria dell'informazione e le forme della computazione da essi implementate, concentrandosi in particolare sul periodo coincidente con la Seconda Guerra Mondiale e sugli sviluppi culturali e sociotecnici immediatamente successivi.

Proprio in quegli anni prese forma, sotto la spinta di contingenze storiche ben definite, una configurazione della computazione e della sua relazione con l'essere umano che non solo trovò una concretizzazione tecnica, ma incise profondamente anche sull'organizzazione della società. Le esigenze belliche, infatti, catalizzarono risorse scientifiche, economiche e politiche, producendo un intreccio fra ricerca militare, sviluppo tecnologico e innovazione industriale che, nel dopoguerra, si tradusse in infrastrutture durature, nuove forme di cooperazione scientifica e in un immaginario culturale destinato a lasciare una traccia materiale e simbolica di lungo periodo.

Senza tornare ulteriormente sullo studio delle armi nucleari e sulla spinta data allo sviluppo del computer dalla ricerca balistica, è sufficiente ricordare che la cibernetica trovò la propria genesi nel progetto di un predittore antiaereo, nel quale Wiener condensò i suoi studi di matematica probabilistica, generalizzandoli in una teoria della retroazione negativa come struttura comune ai sistemi. In modo parallelo - sebbene la prospettiva wieneriana fosse più orientata al contenuto semantico e a una forma "neuromorfica" - la teoria dell'informazione di Shannon, nata dal suo lavoro di matematizzazione della crittoanalisi e sostanzialmente coincidente con essa, si caratterizzò per la totale indifferenza rispetto al contenuto del messaggio e alla natura di mittente e destinatario.

Turing, a sua volta, maturò la convinzione che la sua macchina universale potesse essere realizzata in hardware solo dopo aver conosciuto Flowers e Colossus, macchina frutto di una serie di fortunate coincidenze, la cui segretezza - se non fosse stata mantenuta - avrebbe forse inciso in modo diverso sull'evoluzione del calcolatore. Sullo sfondo si staglia l'idea, ispirata dagli studi di Turing, dell'equivalenza formale e funzionale tra cervello, circuiti e computazione, che trovò un emblema nel lavoro di McCulloch e Pitts. Tale intuizione alimentò la retorica

⁶⁹³ "The 1940s was a decade when fundamental questions were raised about the proper role of how human beings should interact with complex control machinery"..

cibernetica che accompagnò la nascita del calcolatore a programma memorizzato e dell'architettura di von Neumann, divenuta in seguito modello della mente per l'Intelligenza Artificiale classica e per le scienze cognitive, e diffusasi in tutti gli ambiti in cui il computer diveniva via via strumento insostituibile: dalla ricerca scientifica al commercio. Negli anni '50 e '60 essa si impose come principio sistemico, incarnandosi nelle *trading zone* e nei linguaggi ibridi delle simulazioni, fino a configurarsi come modello di organizzazione sociale.

In continuità e in reazione a tale visione, intrecciandosi con sviluppi tecnologici, imprenditoriali e culturali, nacque la computazione personalizzata, ovvero il personal computer. Essa prese forma a partire dall'idea che le macchine potessero diventare strumenti di liberazione ed espansione della mente - concepita, al pari del computer, come elaboratore di informazioni - e che nello spazio digitale si potesse intravedere una nuova frontiera di libertà, assimilabile alle comuni rurali o alle vastità cosmiche da colonizzare. Questo spazio artificiale, la realtà digitale, si aprì simbolicamente con l'implementazione del metodo Monte Carlo sull'ENIAC.

Gli output delle schede perforate, il display di ENIAC o le combinazioni di luci del tubo catodico della macchina di Manchester rendevano visibile la natura digitale dell'informazione o la sua elaborazione interna altrimenti invisibile. Tuttavia, "l'estensione crescente del dominio binario richiese ben presto forme di accesso molto più congeniali alla mente umana rispetto a quelle digitali" [Floridi, 1999, p.14 trad. mia]⁶⁹⁴.

Tra queste, alcune delle tappe fondamentali nel processo di visualizzazione e manipolazione dell'infosfera sono state lo schermo del computer, la VDU (Visual Display Unit); la GUI (Graphic User Interface), l'interfaccia che permette all'utente di interagire con l'artefatto digitale attraverso icone, presentata il 24 gennaio 1984 da Steve Jobs per il Macintosh, ma basata sugli sviluppi realizzati nel decennio precedente allo Xerox *Palo Alto Research Center* (PARC) [Haigh, Ceruzzi, 2021, p. 244], a loro volta fondati sulle ricerche dell'*Augmentation Research Center* (ARC) diretto da Douglas Engelbart presso lo *Stanford Research Institute* (SRI), che negli anni '60 produsse innovazioni decisive nell'interazione anche "corporea" con il calcolatore e lo spazio digitale, come il mouse o le finestre [*windows*]⁶⁹⁵.

Accanto a queste si sono sviluppati approcci come il WIMP (*Windows, Icons, Menu, Pointer*) e il WYSIWYG (*What You See Is What You Get*), termine popolarizzato da Jonathan

⁶⁹⁴ "The growing extent of the binary domain soon required forms of access far more congenial to the human mind than the digital".

⁶⁹⁵ Sul lavoro, le idee, l'impatto e le conseguenze del lavoro di D. Engelbart si rimanda a Rheingold, 2000, Bardini, 2000 e Petrick, 2020.

Seybold (1945–2004), analista e consulente editoriale, tra i pionieri della composizione tipografica assistita da computer, che lo riprese da una battuta del comico afroamericano Flip Wilson-il primo afroamericano ad apparire regolarmente in televisione - per descrivere quelle interfacce in cui il documento veniva visualizzato sullo schermo in modo fedele al risultato finale stampato, con grafica, caratteri e gli altri tipi di formattazione che comparivano direttamente sullo schermo, invece di dover ricorrere ad oscuri comandi che non fornivano alcun riscontro visivo immediato. Nato anch'esso allo Xerox PARC con l'editor *Bravo* (1974), il primo editor-formatter WYSIWYG, il paradigma si diffuse in sistemi come *Star* del 1981, primo sistema commerciale da ufficio con interfaccia grafica sviluppato da Xerox; *LisaWrite* e *MacWrite*, i due programmi di videoscrittura WYSIWYG sviluppati da Apple rispettivamente per il Lisa del 1983 e per il Macintosh del 1984 che contribuirono alla diffusione del paradigma presso il grande pubblico. [Haigh, Ceruzzi, 2021, p.250 e p.270; Myers, 1996].

Le innovazioni tecnologiche, analizzate secondo i vettori di sviluppo che abbiamo preso in esame, hanno rendere possibile, in una sorta di curioso cammino all'inverso rispetto a quanto preso in esame rispetto al passaggio al digitale, nell'interazione con l'informazione uno "spettacolare ritorno dell'analogico". Quest'ultimo va inteso come "come soglia fondamentale tra il macrocosmo binario e intersoggettivo dell'informazione e il microcosmo mentale della comprensione di ciascun individuo. Di conseguenza, siamo passati dall'essere all'interno di un computer reale all'essere all'interno di una realtà computerizzata" [Floridi, 1999, p. 14 trad. mia]⁶⁹⁶. Avevamo lasciato l'analisi sugli sviluppi dell'architettura di von Neumann con una

⁶⁹⁶ "[The previous developments] have made possible a spectacular return of the analogue, as the fundamental threshold between the binary, intersubjective macrocosm of information and the mental microcosm of each individual's understanding. As a result, we have moved from being inside a real computer to being inside a computerised reality". A partire da questo "microcosmo mentale della comprensione di ciascun individuo" [Floridi, 1999, p. 14, trad. mia], è possibile considerare la natura del Sé come entità informazionale in rapporto alle tecnologie computazionali. Queste ultime, come osserva Turkle, possiedono una qualità cangiante che le rende «un mezzo ideale per la costruzione di un'ampia varietà di mondi privati e, attraverso di essi, per l'esplorazione del sé» [«The computer's chameleonlike quality [...] makes it an ideal medium for the construction of a wide variety of private worlds and, through them, for self-exploration», Turkle, 1984, p. 21, trad. mia]. Da una prospettiva informazionale, il sé può essere interpretato secondo due linee principali: un approccio di tipo *lockeano*, che fonda l'identità personale sull'unità della coscienza e la continuità della memoria, oppure un approccio narrativo, che intende l'identità come artefatto sociale, o *autobiografico*, prodotto dalla capacità di raccontarsi e di essere raccontati [Floridi, 2011b; Floridi, 2014, pp.67-69 e pp. 77-78]. In entrambi i casi, la costituzione del sé si realizza attraverso processi di elaborazione informazionale: coscienza, memoria e narrazione sono stati dinamici di informazione. Ma se, come suggerisce l'informazione-shannoniana, ogni ente è in ultima analisi un *pattern* di informazioni, allora sorge un problema cruciale: come può un sé mantenersi coerente e duraturo nel tempo se è costituito da flussi informazionali sempre mutevoli? O meglio: se "l'individualità [...] è quella di una fiamma, di una forma piuttosto che di sostanza" [Wiener, 1950, p. 125, trad. it.] e se ogni ente nell'universo è informazione, che cosa garantisce l'unità del sé, impedendo che esso si dissolva in bit indifferenziati? La mera successione di percezioni o l'accumulo di tracce mnestiche e narrative non bastano da soli a spiegare la coesione. È necessario un principio di organizzazione che stabilizzi il flusso dei bit e li mantenga coerenti. È in questa prospettiva che il modello delle *tre membrane* proposto da Floridi - *corporea, cognitiva e coscienziale* - può rivelarsi utile. L'identità

domanda aperta circa l'appropriatezza dell'impiego della metafora cibernetica in relazione al design seriale dei calcolatori e al rapporto con l'essere umano. Nelle pagine successive, a partire dall'approfondimento di questa analogia funzionale da parte del matematico ungherese, integrata con la teoria dell'informazione, cambieremo il livello di astrazione della nostra indagine, entrando all'interno dei componenti - biologici e artificiali - per mettere in luce la genesi concettuale degli sviluppi computazionali che hanno reso il digitale, e più in generale tecnologie dell'informazione, così pervasivi nella nostra contemporaneità.

personale emerge da un processo di progressivo distacco dall'ambiente, che garantisce stabilità attraverso l'incapsulamento, l'elaborazione e la rappresentazione dei dati. La membrana corporea segna la separazione fisica dall'ambiente, permettendo di resistere all'entropia; la membrana cognitiva introduce la capacità di codificare e comunicare l'informazione, rendendo possibile anche il modello della comunicazione di Shannon; la membrana coscienziale, infine, consente la costruzione di un mondo interno di significati e narrazioni, che rende possibile l'autocoscienza [Floridi, 2011b]. In questo senso, "i sé sono le tecnologie *negentropiche* ultime, attraverso cui l'informazione temporaneamente supera la propria entropia, diventa cosciente, e capace di raccontare la storia della propria emergenza in termini di un progressivo distacco dalla realtà esterna. Ci sono ancora solo strutture informazionali. Ma alcune sono cose, alcune sono organismi, e alcune sono menti, esseri intelligenti e auto-consapevoli. Solo le menti sono capaci di interpretare altre strutture informazionali come cose o organismi o sé. E questa è parte della loro posizione speciale nell'universo" ["Selves are the ultimate negentropic technologies, through which information temporarily overcomes its own entropy, becomes conscious, and is finally able to recount the story of its own emergence in terms of a progressive detachment from external reality. There are still only informational structures. But some are things, some are organisms, and some are minds, intelligent and self-aware beings. Only minds are able to interpret other informational structures as things or organisms or selves. And this is part of their special position in the universe", Floridi, 2011b]. Una dinamica, oggi, ulteriormente amplificata dalle tecnologie digitali e dalle ICT, che rafforzano il carattere egopoietico ed ecologico del sé.

Cap.6: “Johnny Builds Bombs and Johnny Builds Brains”: verso la struttura computazionale del sistema nervoso

1. Microorganismi nell’ENIAC: la cibernetica riformata

Finora abbiamo visto come quella che abbiamo definito *metafora cibernetica* – l’analogia funzionale tra attività logica, cervello e computer – abbia sostenuto l’elaborazione del paradigma del calcolatore a programma memorizzato, l’EDVAC, e la sua successiva diffusione attraverso snodi che hanno generato *trading zone* e linguaggi ibridi su cui essa si è progressivamente incardinata: istanze scientifico-intellettuali, come la *Teleological Society* e le *Macy Conferences*.

Il “circolo cibernetico”, come il gruppo fu chiamato in seguito, fornì negli anni Quaranta contributi sostanziali alla scienza. Analizzarono la nozione di rumore, delimitarono il concetto di informazione e ne dimostrarono il significato termodinamico negentropico, progettaroni filtri capaci di prevedere in modo intelligente i valori futuri di serie temporali stazionarie, scoprirono gli aspetti servomeccanici dell’attività volontaria finalizzata, svilupparono un calcolo per i processi neuronali e una teoria della Gestalt o del riconoscimento dei pattern, e riconobbero la modalità operativa comune del cervello e del computer [Masani, 1990, pp. 239-240].

Parallelamente, queste riflessioni – maturate all’interno di un contesto preciso, quello della ricerca bellica, che ne garantiva la plausibilità – trovarono applicazioni concrete: dal metodo Monte Carlo e la sua implementazione al progetto SAGE, in cui la metafora dell’organismo si trasformava in principio organizzativo esteso a tutti i sistemi, compresi quelli socio-politici. Il “verbo” dell’EDVAC e della sua architettura trovò momenti di diffusione decisivi, come le *Moore School Lectures* dell’estate 1946, che portarono i principi del programma memorizzato a costruttori di calcolatori provenienti da tutto il mondo, mentre i progressi dell’elettronica mettevano a disposizione nuovi strumenti per dare corpo all’implementazione delle funzioni logiche e neurologiche, con i risvolti sociologici e contro-culturali che abbiamo analizzato

A completare questo quadro si aggiungeva la formulazione rigorosa, in diverse sfumature, della teoria dell’informazione, che culminerà nella proposta di Shannon del 1948 [Shannon, 1948]: “Il concetto di messaggio non si limitava ai messaggi di origine umana - vi

rientravano quelli implicati nel lampeggiare delle lucciole e quelli che scorrono attraverso le macchine elettriche. La questione dell'intelligenza, che ricondusse alla logica e alla possibilità, già intravista da Leibniz e Pascal, di automi logici e di intelligenza artificiale, era dunque inclusa" [Masani, 1990, p. 239, trad. mia]⁶⁹⁷.

Abbiamo inoltre messo in luce come la relazione *simbiotica* tra essere umano e calcolatore, originariamente sviluppata per l'esecuzione di specifiche computazioni, abbia condotto alla nascita della programmazione e dei sistemi operativi. Questi ultimi sostituivano l'operatore – o, più spesso, l'operatrice – nell'inserimento manuale delle istruzioni tramite cablaggi o schede perforate, rendendo possibile la memorizzazione e la gestione del programma all'interno della macchina stessa.

Al contempo, sono emerse alcune criticità: il *collo di bottiglia* dell'architettura seriale di von Neumann, che rifletteva – nel prisma tra cognizione, struttura del calcolatore e neurofisiologia – legato ad una precisa idea del cervello come elaboratore di informazioni; i limiti della granularità del modello di astrazione del neurone di McCulloch e Pitts; e la traccia dei limiti della computabilità nell'esaurire le facoltà del cervello (in cui si inserisce quella relazione con l'essere umano al cuore della nozione rigorosa di algoritmo), che portarono Turing a studiare le macchine non organizzate.

È a partire da queste riflessioni che diventa possibile interrogare criticamente l'impiego dell'armamentario concettuale della cibernetica, nelle forme in cui esso si è tradotto concretamente nella computazione e nello studio della cognizione, e al tempo stesso individuare linee di sviluppo alternative dell'elaborazione dell'informazione che hanno condotto ai paradigmi attuali dell'IA.

Un percorso che prende avvio da alcune riflessioni della figura che, forse più di ogni altra, ha rappresentato il punto di connessione tra la dimensione teorica e la realizzazione concreta del calcolatore: John von Neumann.

Nonostante il naufragare del progetto della *Teleological Society*, "von Neumann continuò a riflettere sulla natura dell'elaborazione biologica dell'informazione e a discutere le sue idee con altri scienziati" [Aspray, 1990, p.184 trad. mia]⁶⁹⁸.

⁶⁹⁷ "The concept of message was not restricted to messages of human origin-those involved in the flashing of fireflies and those flowing through electrical machinery were also included. The question of intelligence, which led back to logic, and to the possibility, discerned by Leibniz and Pascal, of logical automata and machine intelligence were thus included".

⁶⁹⁸ "Although plans for the Teleological Society fell apart, von Neumann continued to ponder the character of biological information processing and discuss his ideas with other scientists".

Un esempio significativo del suo pensiero sul rapporto tra organismi intesi come sistemi informativi e il calcolatore è una lunga lettera del 29 novembre 1946 - già richiamata a proposito del legame di mutua influenza tra il matematico ungherese e il lavoro di Turing, e redatta pochi mesi dopo la dimostrazione pubblica dell'ENIAC - indirizzata a Norbert Wiener, nella quale von Neumann delineava le proprie idee in vista di un incontro previsto per il dicembre dello stesso anno.

La lettera si apre, in modo insolito per von Neumann, con un acuto senso di pessimismo. Il matematico ungherese scriveva a Wiener, "penso che tu possa condividere con me il senso di frustrazione che sto cercando di esprimere" [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁶⁹⁹. Questo scoramento era dovuto al fatto che i "risultati di Turing, McCulloch e Pitts [la base per la trattazione computazionale degli organismi] fossero troppo generali per fornire informazioni utili sul funzionamento del meccanismo neurale" [Aspray, 1990, p.184 trad. mia]⁷⁰⁰.

I nostri pensieri - intendo i tuoi, quelli di Pitts e i miei - si sono finora concentrati principalmente sulla neurologia, e più specificamente sul sistema nervoso umano, e in esso soprattutto sul sistema nervoso centrale. Così, nel tentativo di comprendere il funzionamento degli automi e i principi generali che li governano, abbiamo scelto come azione immediata l'oggetto più complicato sotto il sole - letteralmente [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰¹.

Nonostante alcuni successi iniziali dovuti allo sforzo di Pitts e McCulloch, Wiener, Rosenblueth e, quelli, nuovamente sottolineati, di Turing, sull'argomento, il quale "ha fornito informazioni molto interessanti [...]" E grazie al quale, prosegue von Neumann, "il nostro pensiero - o almeno il mio - sull'intero argomento degli automi sarebbe molto più modesto di quanto sia, se questi tentativi estremamente audaci [...]. Tuttavia, penso che questi successi non debbano accecarci di fronte alle difficoltà dell'argomento" [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰².

⁶⁹⁹ "I think that you will feel with me the force of the frustration that I am trying to express".

⁷⁰⁰ "It argued that the results of Turing, McCulloch, and Pitts were too general to be informative about the working of the neural mechanism".

⁷⁰¹ "Our thoughts - I mean yours and Pitts' and mine - were so far mainly focused on the subject of neurology, and more specifically on the human nervous system, and there primarily on the central nervous system. Thus, in trying to understand the function of automata and the general principles governing them, we selected for prompt action the most complicated object under the sun - literally".

⁷⁰² "In spite of its formidable complexity this subject has yielded very interesting information under the pressure of the efforts of Pitts and McCulloch, Pitts [*sic*], Wiener and Rosenblueth. Our thinking - or at any rate mine - on the entire subject of automata would be much more modest than it is, if these extremely bold efforts - with which I

Il sistema nervoso è un'entità troppo complessa, secondo von Neumann, per essere studiata secondo i termini al momento impiegati. "Le difficoltà sono quasi troppo ovvie per essere menzionate: risiedono nell'eccezionale complessità del sistema nervoso umano, e in effetti di qualsiasi sistema nervoso". Nonostante i primi successi del "grande e positivo contributo di Turing - insieme a Pitts e McCulloch", i quali hanno dimostrato "in termini assoluti e con onesta generalità, che qualsiasi cosa di natura brouweriana può essere realizzata da un meccanismo appropriato, e specificamente da un meccanismo neurale", l'esito non fa sperare per il meglio: "la situazione appare piuttosto peggiore che migliore rispetto a prima" [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰³.

La complessità del cervello, e in generale di ogni apparato neurofisiologico, appariva soverchiante sia rispetto alla trattazione computazionale adottata dai "cibernetici", sia rispetto ai metodi sperimentali che lo stato dell'arte della seconda metà degli anni Quaranta rendeva disponibili, soprattutto se confrontata con quello che allora si considerava l'analogo funzionale dell'apparato cerebrale umano: l'ENIAC, nel frattempo in fase di conversione al *stored program principle*.

Le parole di von Neumann rendono le difficoltà in questo senso:

Per comprendere il cervello con i metodi sperimentali oggi a nostra disposizione, bisognerebbe trattare l'ENIAC senza alcun intervento più fine che far scorrere qualcosa attraverso gli organi critici, con nessun metodo di intervento più delicato che giocare con una manichetta antincendio (anche se la si riempisse di cherosene o di nitroglicerina invece che d'acqua) o gettare ciottoli nel circuito [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰⁴.

Il matematico ungherese aggiunge inoltre - con una considerazione di notevole rilevanza per il nostro discorso e su cui torneremo - che il sistema assunto come modello per gli studi

would like to put us, for one part, the very un-neurological theses of A. Turing - had not been made. Yet, I think that these successes should not blind us to the difficulties of the subject, difficulties, which, I think, stand out now just as - if not more - forbiddingly as ever".

⁷⁰³ "The difficulties are almost too obvious to mention: They reside in the exceptional complexity of the human nervous system, and indeed of any nervous system. What seems worth emphasizing to me is, however, that after the great and positive contribution of Turing - cum Pitts and McCulloch - is assimilated, the situation is rather worse than better than before. Indeed, these authors have demonstrated in absolute and honest generality, that anything and everything Brouwerian can be done by an appropriate mechanism, and specifically by a neural mechanism - and that even one definite mechanism can be "loaded" against the argument".

⁷⁰⁴ "To understand the brain with the experimental methods at our disposal now, one would have to treat the ENIAC with no more intervention than playing across the critical organs, with no method of intervention more delicate than playing with a fire hose (although one might fill it with kerosene or nitroglycerine instead of water) or dropping cobblestones into the circuit".

sulla computazione e sull'elaborazione dell'informazione non è neppure interamente di natura digitale.

Inoltre, il sistema non è nemmeno puramente digitale (cioè neurale): è intimamente connesso a un sistema analogico (cioè umorale o ormonale) estremamente complesso, e quasi ogni circuito di retroazione attraversa entrambi i settori, se non addirittura il "mondo esterno" (cioè il mondo al di fuori dell'epidermide o all'interno dell'apparato digerente). E contiene, nella sua parte digitale, un numero di unità un milione di volte superiore a quello dell'ENIAC. E le nostre possibilità intellettuali nei suoi confronti sono approssimativamente pari a quelle di qualcuno nei confronti dell'ENIAC, se non ha mai sentito parlare di alcun ramo dell'aritmetica [von Neumann,1946, lettera a Wiener 29 novembre trad. mia]⁷⁰⁵.

Il pessimismo che von Neumann manifesta a Wiener nella sua missiva programmatica non si attenua nemmeno di fronte agli approcci alternativi a quelli appena criticati. A suo giudizio, infatti, né la psicologia della Gestalt né altre metodologie di tipo verbale offrono prospettive scientificamente feconde: "Il sistema nervoso centrale è complicato e, perciò, ha pieno diritto di possedere attributi e caratteristiche altrettanto complicati. Non lasciamo che la nostra facile familiarità con esso, mediata dalla coscienza soggettiva, ci induca in alcuna illusione in questo senso" [von Neumann,1946, lettera a Wiener 29 novembre trad. mia]⁷⁰⁶.

Né tantomeno sarebbe scientificamente proficuo avere per oggetto d'indagine sistemi nervosi meno complicati di quello umano, come ad esempio quello delle formiche, in cui il rapporto di complessità, in quanto a numero di neuroni, scende da 10^{10} a 10^6 , poiché "perdiamo quasi tanto quanto guadagniamo. Man mano che la parte digitale (neurale) si semplifica, la parte analogica (umorale) diventa meno accessibile, i malfunzionamenti tipici meno noti, l'argomento meno articolato, e le nostre possibilità di comunicare con esso si impoveriscono sempre di più nei contenuti" [von Neumann,1946, lettera a Wiener 29

⁷⁰⁵ "Besides the system is not even purely digital (i.e. neural): It is intimately connected to a very complex analogy (i.e. humoral or hormonal) system, and almost every feed-back loop goes through both sectors, if not through the "outside" world (i.e. the world outside the epidermis or within the digestive system) as well. And it contains, even in its digital part, a million times more units than the ENIAC. And our intellectual possibilities relatively to it are about as good as somebodies vis-a-vis the ENIAC, if he has never heard of any part of arithmetic. It is true that we know a little about the syndromes of a few selected breakdowns-but that is not much".

⁷⁰⁶ "Further: I doubt that the "Gestalt" theory, or anybodies verbal theory will help any. The central nervous system is complicated, and therefore its attributes and characteristics have every right to be complicated. Let not our facile familiarity with it, through the medium of the subjective consciousness, fool us into any illusions in this respect".

novembre trad. mia]⁷⁰⁷. Ed è, inoltre, persino poco utile considerare la *cellula neuronale* come analogo dei “componenti standard” del calcolatore elettronico.

È un errore argomentare che, poiché il neurone è una cellula (e, di fatto, parte del suo involucro isolante individuale è multicellulare), dobbiamo considerare soltanto organismi pluricellulari. La cellula è chiaramente un eccellente “componente standard”, molto flessibile e adatta a differenziarsi sia nella forma sia nella funzione, e gli organismi superiori ne fanno ampio uso. Ma la sua capacità di autoriprodursi indica che essa possiede in sé alcuni degli attributi decisivi degli organismi integrati - e alcune cellule (per esempio i leucociti) sono esseri completi e autonomi. [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰⁸.

Un’assiomatizzazione del sistema nervoso, argomenta von Neumann, non dovrebbe avere come suoi elementi primitivi le cellule (o la loro modellizzazione concettuale) proprio in virtù di una differenza “evolutiva” essenziale con i componenti standard elettronici. Questi sono già, nel loro ordine di complessità, “individualizzati, adatti alla produzione di massa e, nonostante il loro carattere “standard”, idonei a una differenziazione finalizzata. Questo è chiaramente uno stile tardivo, altamente sviluppato, e non quello ideale per il primo approccio di un estraneo alla materia, in un tentativo di comprensione” [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷⁰⁹.

Tuttavia, malgrado lo scoramento dimostrato nella prima parte della lettera a Wiener, la “seconda metà della lettera era più ottimista; von Neumann suggerì un programma di ricerca empirica che prevedeva lo studio del batteriofago” [Aspray, 1990, p. 185, trad. mia]⁷¹⁰. Si tratta di organismi “sub-cellulari”, analoghi ai virus - entità di quella *twilight zone* tra la vita e la morte,

⁷⁰⁷ “Next: If we go to lower organisms from man with 10^{10} neurons to ants with 10^6 , neurons-we lose nearly as much as we gain. As the digital (neural) part simplifies, the analogy (humoral) part gets less accessible, the typical malfunctions less known, the subject less articulate, and our possibilities of communicating with it poorer and poorer in content”.

⁷⁰⁸ “It is a fallacy, if one argues, that because the neuron is a cell (indeed part of its individual insulating wrapping is multicellular), we must consider multicellular organisms only. The cell is clearly an excellent “standard component”, highly flexible and suited to differentiation in form and in function, and the higher organisms use it freely. But its self-reproductivity indicates that it has in itself some of the decisive attributes of the integrated organisms-and some cells (e.g. the leukocytes) are self-contained, complete beings”.

⁷⁰⁹ “This in itself should make one suspicious in selecting the cells as the basic “undefined” concepts of an axiomatism. To be more *par terre*: Consider, in any field of technology, the state of affairs which is characterized by the development of highly complex “standard components”, which are at the same time individualized, well suited to mass production, and (in spite of their “standard” character) well suited to purposive differentiation. This is clearly a late, highly developed style, and not the ideal one for a first approach of an outsider to the subject, for an effort towards understanding”.

⁷¹⁰ “The second half of the letter was more upbeat; von Neumann suggested a program of empirical research involving the study of the bacteriophage”.

tra l'organico e l'inorganico, che affascinava Wiener [Cfr. Masani, 1990, pp. 248-249] - i quali rappresentano uno stadio arcaico di sviluppo, in cui la differenziazione dei componenti standard non si è ancora manifestata, e possono costituire oggetto di analisi per lo studio degli organismi biologici dal punto di vista computazionale.

Gli organismi sub-cellulari del tipo dei virus o dei batteriofagi possiedono i tratti decisivi di qualsiasi organismo vivente. Sono autoriproduttivi e sono in grado di orientarsi in un ambiente non organizzato, di muoversi verso il nutrimento, di appropriarsene e di utilizzarlo. Di conseguenza, una "vera" comprensione di questi organismi può costituire il primo passo rilevante in avanti e, possibilmente, il più grande passo che possa essere richiesto [von Neumann, 1946, lettera a Wiener 29 novembre trad. mia]⁷¹¹.

Come, a ragione, nota P. R. Masani, la missiva di von Neumann a Wiener presenta una profondità dialettica "penetrante e convincente". Il successo (ad esempio, del modello di McCulloch-Pitts) - da lui stesso denunciato come solo apparente nella prima parte della lettera "dell'assiomatizzazione a grana grossa delle reti nervose esclude l'*experimentum crucis* necessario per andare più a fondo, e rende indispensabile lo studio a un livello citologico microscopico più fine". Tuttavia, pur mettendo da parte la complessità intrattabile del sistema nervoso umano, non soltanto un sistema più semplice come quello della formica, ma persino la cellula stessa "è eccessivamente complessa, in virtù della struttura acquisita durante milioni di anni di evoluzione. Era necessario qualcosa di più primitivo e formativo: un organismo subcellulare". [Masani, 1990, p.248 trad. mia]⁷¹². Solo grazie allo studio di questi "sistemi semplici", si poteva arrivare alla comprensione della natura informazionale del sistema nervoso umano e trattarne la complessità [Copeland, Fan 2023 trad. mia]⁷¹³.

Nella lettera a Wiener, von Neumann tratteggia le possibilità di un metodo scientifico per lo studio di queste entità fondamentali. Egli intravede innanzitutto un approccio empirico,

⁷¹¹. Now the less-than-cellular organisms of the virus or bacteriophage type do possess the decisive traits of any living organism: They are self-reproductive and they are able to orient themselves in an unorganized milieu, to move towards food, to appropriate it and to use it. Consequently, a "true" understanding of these organisms may be the first relevant step forward and possibly the greatest step that may at all be required.

⁷¹² The dialectic in von Neumann's letter is both penetrating and compelling. The very success of the coarse-grain axiomatization of nervous nets precludes the *experimentum crucis* needed to go deeper, and necessitates study at the finer microscopic cytological level. But even the ant with its 106 neurons is too complex, let alone the human being with 1010 neurons. Indeed, even the simplest cell is over-complex by virtue of the structure acquired during millions of years of evolution. Something more primitive and formative was needed: a subcellular organism.

⁷¹³ On 29 November 1946, von Neumann had written to Wiener, proposing as a prelude to the study of the human nervous system, with its 'exceptional complexity', a study of 'simpler systems', such as viruses and bacteriophages.

basato sulla combinazione tra raccolta sistematica di dati e sviluppo mirato di strumenti e tecniche di osservazione.

La strategia di studio sperimentale comprende più filoni: raccogliere le principali evidenze sulla fisiologia di virus e batteriofagi e sul rapporto gene–enzima; acquisire una conoscenza solida delle opinioni e dei risultati attuali riguardo alla struttura delle proteine (denunciata come carente nel gruppo dei “cibernetici”); padroneggiare i principi della determinazione della struttura chimico–organica tramite analisi ai raggi X e trasformate di Fourier, supportate da manipolazioni chimiche; e infine esplorare e perfezionare le metodologie di microscopia elettronica, spingendosi in un futuro ragionevole verso la futura ottica a protoni per ottenere risoluzioni prossime alla scala atomica, con possibili ricadute potenziali straordinarie in fisiologia⁷¹⁴ [Cfr. von Neumann, lettera a Wiener 29 novembre 1946]⁷¹⁵

⁷¹⁴ È nello sviluppo della microscopia che von Neumann intravede gli sviluppi più interessanti [“I suspect that the main possibilities may well lie in that direction]. Von Neumann nota che, al momento in cui scrive, la risoluzione dei microscopi elettronici magnetici è di circa 10 Å (ångström, corrispondente a 0,1 nanometri), insufficiente ma vicina al necessario per distinguere distanze atomiche, richiedendo solo un miglioramento di 10–20 volte. Ma, il matematico prevede che “il microscopio a protoni potrebbe essere disponibile già tra 2–4 anni, e supererebbe certamente queste difficoltà” [On the other hand, the proton microscope need not be more than 2–4 years in the future, and it would certainly overcome these difficulties]. Le conseguenze di questo sviluppo tecnologico entusiasmano von Neumann, che scrive: “Oltre a ciò, tutti questi sviluppi potrebbero essere accelerati. Naturalmente, tutti sanno cosa significherebbe una risoluzione di 1 Å: si potrebbe “vedere” un atomo di idrogeno, e con un po’ di più, diciamo 1/5 Å, si potrebbe “vedere” la nube di carica di Schrödinger degli elettroni orbitali. Ma le implicazioni fisiologiche sarebbero ancora più straordinarie, e dovrebbero ricevere grande attenzione nell’immediato futuro” [Of course, everybody knows what a 1–1/2 Å resolution would mean: One could “look” at an H atom, and with a little more, say 1/5 Å, one could “see” the Schroedinger-charge- cloud-of the orbital electrons. But the physiological implications are even more extraordinary, and they should receive a great deal of emphasis in the immediate future, von Neumann, lettera a Wiener 29 novembre 1946 trad. mia]. Una intuizione di von Neumann presente nella lettera a Wiener si rivelerà carica di futuro (e può essere messa in parallelo al trattamento algebrico della genetica della tesi di dottorato di Shannon che abbiamo analizzato). Il suggerimento di investigare “la fisiologia di virus e batteriofagi, e tutto ciò che si sa sul rapporto gene–enzima” si dimostrerà carico di futuro, sebbene non sviluppato da membri della cerchia dei cibernetici. Nel 1952 Alfred Hershey (1908–1997), genetista, e Martha Chase (1927–2003), dimostrarono che il codice ereditario è contenuto nell’acido nucleico del fago e non nel suo involucro proteico. Nel 1955 il biochimico Heinz Frankel-Conrat (1910–1999), mostrò che è proprio un enzima presente sul rivestimento del fago a rompere la parete batterica [Cfr. Masani, 1990, p.248; Israel, Millán Gasca, 2008, pp.223–224]. Successivamente, l’acido nucleico del fago fu identificato come DNA, la cui struttura era stata scoperta nel 1953 da James D. Watson (1928–vive) e Francis Crick (1916–2004), basandosi anche sul lavoro intorno alla diffrazione a raggi X di Rosalind Franklin (1920–1958). Il *gene egoista* di Richard Dawkins (1941–vive) (Dawkins, 1976) rappresenta un esempio classico di DNA come codice genetico in senso informativo. Questa prospettiva si accorda con l’idea di von Neumann del DNA come “programma” di istruzioni evolutive, interpretabile anche attraverso la teoria dei giochi, che formalizza strategie e processi riproduttivi in termini competitivi e cooperativi.

⁷¹⁵ “At any rate, I think that we could do these things: Study the main types of evidence: Physiology of viruses and bacteriophages, and all that is known about the gene-enzyme relationship. (Genes are probably much like viruses and phages, except that all the evidence concerning them is indirect, and that we can neither isolate them nor multiply them at will.) Try to learn a reasonable amount about the present state of knowledge and opinions concerning protein structure. Study the methods of organic-chemical structure determination by x-ray analysis and Fourier-analysis with their necessary complement of manipulations. Study the principles and methods of

Il batteriofago, in questo quadro, viene presentato come un esempio ideale di organismo estremamente semplice ma completo: una fase parassitaria del *Bacillus coli*, con morfologia riconoscibile (testa e coda) e dimensioni estremamente ridotte - nell'ordine di pochi centesimi di micron ($1/15 \times 1/60 \mu \times 1/25 \mu$) - che consentono di calcolarne volume, massa e numero di atomi costituenti. Può essere moltiplicato a piacere e "contato" con la stessa affidabilità con cui un contatore Geiger rileva le particelle elementari. Le sue membrane, di natura monomolecolare, "svolgono la loro funzione in modo altamente meccanico" - ad esempio nella "permeabilità attiva", dove ioni vengono fisicamente spinti attraverso la membrana con un dispendio di energia metabolica. [Cfr. von Neumann, lettera a Wiener 29 novembre 1946]⁷¹⁶.

Le considerazioni intorno a questi organismi "base" sub-cellulari prendono le mosse nel pensiero di von Neumann dalle ricerche del biofisico tedesco-statunitense Max Delbrück (1906-1981)⁷¹⁷. Era intenzione di von Neumann il voler includere nel gruppo interdisciplinare delle *Macy Conferences* anche un genetista come membro regolare, e Max Delbrück era considerato il candidato ideale. Delbrück, tuttavia, partecipò solo al quinto incontro, svolto il 18-19 marzo 1948, e non tornò più [Cfr. Heims, 1991, pp.106-110 trad. it.] "Il gruppo interdisciplinare di cibernetica, al contrario, aveva interessi indefiniti ai quali Delbrück si sentiva estraneo". In seguito, come riporta S.J. Heims, commentando l'incontro delle *Macy* a cui aveva partecipato: "Fu estremamente e positivamente vago. La genetica non vi trovò, e all'epoca non vi poteva trovare, alcuno spazio. Io stesso, allora, non ero troppo interessato all'elaborazione delle informazioni nel sistema nervoso centrale dei vertebrati, o in sistemi nervosi più semplici e al suo analogo nei computer, e non lo sono stato neanche in seguito" [Heims, 1991, p.109 trad. it.]. Così, nonostante l'intenzione di von Neumann, il gruppo

electron-microscopy, both in the direction of electron optics and in the direction of object-manipulation. Try to get oriented to the possibilities of proton-microscopy".

⁷¹⁶ "A typical bacteriophage, which can be multiplied at will (and hence "counted" - by the colonies it forms on a suitable substrate-as reliably as elementary particles can be "counted" by a Geiger-counter) [...] It is definitely an animal, with something like a head and a tail. Its dimensions are, I think, $60m \times 25 m \times 25 m$, i.e. its volume is $60 \times 25 \times 25 \times 10^{-21} cm^3 = 3.7 \times 10^{-17} cm^3$. The density may be taken to be 1, hence its mass is about $3.7 \times 10^{-17} gr$, ie. the same as about $2.5 \times 10^7 H$ atoms. Since the average chemical composition of these things is usually about C or N or O per one or two H, the average atomic weight of its constituents is about 6.[...] Furthermore, it is known from the behavior of physiological membranes, that they are monomolecular-or oligomolecular [...] layers, which exercise their function in a highly mechanical way. E.g. the so called "active permeability": The peculiar ability to "permit" ions to pass through the membrane against an electrical field-an activity which clearly must, and demonstrably does, require an energy supply from the metabolism-and which is therefore better described as "pushing the ions across": than as "permitting them to pass" [...] Thus one can really talk of "mechanical elements", each of which may comprise 10 atoms or more".

⁷¹⁷ "A typical bacteriophage [...] is a phage that is parasitic, I think, on the *Bacillus Coli*. It has been extensively worked with, e. g. by Delbrueck at Vanderbilt."

cibernetico, sebbene affascinato dalle idee su vita ed entropia di Erwin Schrödinger, non ebbe mai un genetista o un biochimico come membro regolare.

La lettera di von Neumann a Wiener contiene, inoltre, un'ulteriore indicazione del metodo per trattare queste entità "autoriproduttive": "Posso formulare il problema in modo rigoroso, più o meno nello stile in cui Turing lo fece per il suo meccanismo" [von Neumann, lettera a Wiener 29 novembre 1946]⁷¹⁸.

La considerazione finale di von Neumann segna l'inizio del percorso che lo condurrà a elaborare il concetto di *automa*, destinato a diventare centrale nella chiarificazione del rapporto tra calcolatore e cervello nel suo pensiero. Il suo approccio consisteva nell'affrontare la complessità del sistema nervoso da una prospettiva informazionale, adottando come chiave interpretativa la lente logico-matematica offerta dall'impostazione computazionale di Turing.

"Il tema della complessità - o «complicazione», per dirla con von Neumann - degli automi veniva affrontato ricorrendo al concetto di macchina di Turing, una macchina astratta, capace di stampare e cancellare degli 0 e degli 1 su un nastro di lunghezza infinita e di simulare così qualsiasi processo descrivibile mediante un numero finito di operazioni logiche" [Israel, Millán Gasca, 2008, p.225].

Lo schema generale - che rappresenta un'ulteriore conferma, da parte di von Neumann, dell'importanza del lavoro di Turing - era già stato delineato nella missiva a Wiener: "Alcuni tratti del rapporto gene-enzima [...] sembrano mettere in rilievo alcune varianti della capacità di autoriproduzione, che si sarebbe indotti a investigare anche su basi puramente combinatorie. Vale a dire, l'autoriproduzione può essere simbolizzata dallo schema $A \rightarrow A$. E che dire di schemi come $A \rightarrow B \rightarrow C \rightarrow A$, o $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow D \rightarrow E \rightarrow C$ " [von Neumann, lettera a Wiener, 29 novembre 1946]⁷¹⁹

In questo quadro "cibernetico" di analisi della complessità attraverso lo studio comparato della dimensione biologica e artificiale, si può trovare un antecedente indiretto alle riflessioni di von Neumann nello studio funzionale degli organismi "inferiori" (*lower*) condotto dallo zoologo statunitense Herbert Spencer Jennings (1868-1947).

⁷¹⁸ "I did think a good deal about self-reproductive mechanisms. I can formulate the problem rigourously, in which Turing did it for his mechanisms. I can show that they exist in this system of concepts".

⁷¹⁹ "Certain traits of the gene-enzyme relationship, of the behavior of some mutants, as well as some other phenomena, seem to emphasize some variants of self-reproductivity, which one would be led to investigate on purely combinatorial grounds as well. E.g.: Self-reproductivity may be symbolized by the schema $A \rightarrow A$. What about schemata like $A \rightarrow B \rightarrow C \rightarrow A$, or $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow D \rightarrow E \rightarrow C$."

Jennings mostrò come “organismi e macchine [...], possono essere studiati come organizzazioni funzionali” [Cordeschi, 1998, p. 40], rifiutando l'idea di considerare i microrganismi semplici masse di materiale, il cui comportamento potesse essere chiarito da mere considerazioni chimico-fisiche, e descrivendoli invece come “conformazioni” [*arrangements*] o «strutture» [*structures*] paragonabili a «macchine complesse» [Cordeschi, 1998, p. 40; Jennings, 1910]. Nel suo *The Behavior of the Lower Organisms* [Jennings, 1906], analizzò lo *Stentor* - un protozoo ciliato simile al paramecio - che, minacciato da un agente perturbante, attua in sequenza diversi movimenti fino a selezionare quello efficace al ripristino delle condizioni vitali [Cfr. Cordeschi, 1998, pp.32-34; pp.38-39; Jennings, 1906, p.103 fig.70]. Lo studio funzionale di questi microrganismi consentiva di trattare il sistema come analizzabile, in forma embrionale, secondo la possibilità di una trattazione formale e rigorosa, quale sarà poi anticipata da von Neumann nella sua lettera a Wiener. Non a caso, il modello dello *Stentor* fu assunto da Ross Ashby come esempio di sistema “stabile”, destinato a ispirare la costruzione del suo omeostato [Ashby, 1952; Cfr. Cordeschi, 1998, p.42].

2. Una logica degli errori: dagli automi alla vita

Il lavoro intorno agli *automata* di von Neumann può essere fatto risalire al 1946 in cui elaborò le idee che espose due anni dopo nel già citato simposio *Cerebral Mechanisms in Behavior* del settembre 1948 presso il California Institute of Technology di Pasadena e sponsorizzata dallo Hixon Fund. L'obiettivo del Simposio “era quello di riunire un piccolo gruppo di scienziati di spicco per discutere i meccanismi cerebrali dal punto di vista comportamentale, in contrapposizione a quello biochimico o genetico” [Aspray, 1990, p.189 trad. mia]⁷²⁰. McCulloch, invitato, come detto, alla conferenza, invitò von Neumann a tenere un discorso intorno al concetto di “evoluzione di ordine dal caos”. Ma l'argomento scelto dal matematico ungherese fu diverso: parlò “degli elementi logici di commutazione e della loro integrazione nelle macchine digitali (sia biologiche sia artificiali)” [Aspray, 1990, p.189 trad. mia]⁷²¹.

⁷²⁰ “The objective of the Hixon Symposium was to assemble a small group of distinguished scientists to discuss cerebral mechanisms from the behavioral, as opposed to the biochemical or genetic, viewpoint”.

⁷²¹ “McCulloch, who was also an invited lecturer, encouraged von Neumann to speak on “the evolution of order out of chaos,” but instead von Neumann took as his subject an analysis of logical switching elements and their integration into digital machines (both biological and artificial)”.

È da sottolineare, come fa il biografo di von Neumann, il matematico, filosofo e storico della scienza William Aspray, come le “ripetute incursioni nell’elaborazione dell’informazione biologica e nello studio interdisciplinare della cibernetica sono state ignorate nei precedenti resoconti dell’attività di von Neumann nell’informatica, eppure hanno chiaramente plasmato le sue idee” [Aspray, 1990, p.189 trad. mia]⁷²².

Grazie all’intervento dello psicologo Karl Lashley, che sferrò durante la conferenza un deciso attacco contro il behaviourismo dominante allora in psicologia, trattando dell’ordine seriale nel comportamento, von Neumann trascrisse il suo intervento per la pubblicazione, dal titolo *The General and Logical Theory of Automata* [von Neumann, 1951]. [Cfr. Boden, 2006, p.1269; Gardner, 1985, pp.24-27 trad. it.; Lashley, 1951]. Questo è il “è il primo resoconto pubblicato della teoria degli automi di von Neumann” [Aspray, 1990, p.189 trad. mia]⁷²³.

L’interesse di von Neumann era sintomatico del suo interesse incrociato per la dimensione biologica ed artificiale dei sistemi di processo dell’informazione e sperava che la sua analisi gettasse luce sia sul sistema nervoso umano e sia sull’implementazione di macchine di calcolo sempre più perfezionate. Altrettanto palese, pur nell’assenza di una definizione esplicita di automa, è l’attenzione per concetti “cibernetici” già incontrati: l’autoregolazione, l’equilibrio omeostatico, l’indifferenza funzionale tra sistemi di computo dell’informazione:

Von Neumann non definì esplicitamente il termine “automi”, ma dal suo uso possiamo dedurre l’applicazione a qualsiasi sistema che elabora informazioni come parte di un meccanismo autoregolante. I suoi principali esempi erano il sistema nervoso centrale umano e il computer, ma le sue osservazioni lasciano intendere che considerasse come esempi di automi anche altri sistemi omeostatici presenti negli organismi biologici, così come i sistemi radar, i sistemi telefonici e altri sistemi di comunicazione. Ciò a cui pensava è simile a ciò che oggi è noto come sistema di controllo [Aspray, 1990, pp.189-190 trad. mia]⁷²⁴.

⁷²² “These repeated excursions into biological information processing and the interdisciplinary study of cybernetics have been ignored in previous accounts of von Neumann's computing, yet they clearly shaped his ideas”.

⁷²³ “It is the earliest published account of von Neumann's theory of automata”.

⁷²⁴ “Von Neumann did not explicitly define “automata,” but from his usage we can infer its application to any system that processes information as part of a self-regulating mechanism. His prime examples were the human central nervous system and the computer, but his remarks suggest that he regarded other homeostatic systems within biological organisms, as well as radar systems and telephone and other communication systems, as examples of automata. What he had in mind is akin to what is known today as a control system”.

Malgrado l'assenza di una determinazione precisa del concetto, von Neumann parte dall'importanza, scientifica e metodologica, sempre maggiore che gli automi hanno via via assunto: "Gli automi hanno assunto un ruolo continuamente crescente, e ormai molto considerevole, nelle scienze naturali [...]. Il loro ruolo in matematica presenta un interessante contrappunto a certi aspetti funzionali dell'organizzazione in natura" [von Neumann, 1951, p.1 trad. mia]⁷²⁵

Sono presi in esame due tipologie di automi, gli organismi naturali e quelli artificiali. I primi risultano "molto più complessi e sottili, e quindi molto meno compresi nei dettagli, rispetto agli automi artificiali" [von Neumann, 1951, pp.1-2]⁷²⁶. Malgrado tale differenza nell'ordine di complessità, von Neumann, fornisce una prima valutazione quantitativa degli elementi dei due tipi di automi. Egli osservava che le macchine di calcolo veloce dispongono di almeno 10^4 elementi di commutazione di base e svolgono abitualmente operazioni comprendenti circa 10^6 moltiplicazioni (pari a 10^9 operazioni binarie), delle quali stimava che almeno il 30% fosse cruciale per ottenere un risultato corretto. Valutava inoltre - con una buona approssimazione - che il sistema nervoso centrale umano contasse all'incirca 10^{10} elementi di commutazione di base, ossia neuroni [Cfr. Aspray, 1990, p.190].

Nonostante questa differenza "alcune regolarità che osserviamo nell'organizzazione dei primi possono risultare piuttosto istruttive per il nostro pensiero e la nostra pianificazione riguardo ai secondi; e, viceversa, una buona parte delle nostre esperienze e difficoltà con i nostri automi artificiali può, in una certa misura, essere proiettata nelle nostre interpretazioni degli organismi naturali" [von Neumann, 1951, p.2 trad.mia]⁷²⁷

L'approccio metodologico indicato da von Neumann è quello di suddividere il problema in due parti: "gli organismi possono essere considerati come composti da parti che, fino a un certo punto, sono indipendenti, unità elementari". La prima parte del problema sarà analizzare "la struttura e il funzionamento di tali unità elementari singolarmente". La seconda, invece, "consiste nel capire come questi elementi siano organizzati in un tutto e come il funzionamento del tutto si esprima in termini di questi elementi" La prima parte spetta alla fisiologia, alla fisica

⁷²⁵ "Automata have been playing a continuously increasing, and have by now attained a very considerable, role in the natural sciences. This is a process that has been going on for several decades. During the last part of this period automata have begun to invade certain parts of mathematics too-particularly, but not exclusively, mathematical physics or applied mathematics".

⁷²⁶ "Natural organisms are, as a rule, much more complicated and subtle, and therefore much less well understood in detail, than are artificial automata"

⁷²⁷ "Nevertheless, some regularities which we observe in the organization of the former may be quite instructive - our thinking and planning of the latter; and conversely, a good deal of our experiences and difficulties with our artificial automata can be to some extent projected on our interpretations of natural organisms".

e alla biochimica, mentre la seconda, riguardante la logica matematica, analizzerà il modo in cui i sistemi complessi si organizzano [von Neumann, 1951. p.2 trad. mia]⁷²⁸

Per “liquidare” il primo problema e passare all’analisi del secondo, che costituisce la teoria degli automi vera e propria, von Neumann “ricorse all’assiomatica” [Aspray, 1990, p.191 trad. mia]⁷²⁹. Gli elementi di commutazione, naturali o artificiali, neuroni o tubi a vuoto, vengono trattati – e qui è palese l’approccio cibernetico al problema - come *black box*, considerati come” automatismi, la cui struttura interna non è necessario rivelare, ma che si presume reagiscano a certi stimoli inequivocabilmente definiti con certe risposte altrettanto inequivocabilmente definite”. A partire da questi e dalle loro connessioni è possibile indagare la struttura, la regolarità e il comportamento di quelli più grandi [von Neumann, 1951, p.7 trad. mia]⁷³⁰. La verifica degli assiomi, tuttavia, spetterà sempre alle scienze fisiche e biologiche.

Dopo aver discusso le principali differenze tra automi artificiali - i calcolatori - fondati sul principio analogico e su quello digitale, sottolineando come in questi ultimi il vantaggio consista nella possibilità “sempre più agevole” di “ridurre il rumore computazionale” [von Neumann, 1951, pp.8-9], von Neumann, pur con le cautele metodologiche del caso, riconosceva che gli elementi presi in esame non erano puramente digitali.

I neuroni, ad esempio, manifestano fenomeni come affaticamento, sommazione, auto-oscillazione e stimolazione subliminale; mentre le valvole termoioniche (*vacuum tubes*) possono presentare comportamenti continui o non lineari, legati alle proprietà fisiche dei materiali. Tuttavia, pur accettando questa componente “analogica”, egli trattava entrambi - neuroni e valvole - come organi elettrici digitali, ossia dispositivi che, in condizioni normali, operano secondo il principio del “tutto-o-nulla”, trasmettendo stimoli e risposte mediante impulsi elettrici [Cfr. von Neumann, 1951, pp.9-10, Aspray, 1990, p.191].

Tuttavia, portando oltre il confronto sulla natura materiale degli elementi biologici e artificiali, von Neumann evidenzia come le dimensioni, la spaziatura e le proprietà dei materiali rendessero i calcolatori inevitabilmente meno compatti e complessi degli organismi naturali.

⁷²⁸ “The natural systems are of enormous complexity, and it is clearly necessary to subdivide the problem that they represent into several parts. One method of subdivision, which is particularly significant in the present context, is this: The organisms can be viewed as made up of parts which to a certain extent are independent, elementary units. We may, therefore, to this extent, view as the first part of the problem the structure and functioning of such elementary units individually. The second part of the problem consists of understanding how these elements are organized into a whole, and how the functioning of the whole is expressed in terms of these elements”.

⁷²⁹ “In order to dispose of the first type of study and get on to the theory of automata, von Neumann resorted to axiomatics”.

⁷³⁰ “They have been regarded as 'black boxes.' They are viewed as automatisms, the inner structure of which need not be disclosed, but which are assumed to react to certain unambiguously defined stimuli by certain unambiguously defined responses”.

Dal punto di vista fisico, è nelle caratteristiche materiali naturali che va individuata la superiorità degli automi biologici. “È dunque l’inferiorità dei nostri materiali, rispetto a quelli utilizzati in natura, che ci impedisce di raggiungere l’elevato grado di complessità e le ridotte dimensioni ottenute dagli organismi naturali” [von Neumann, 1951, p.15 trad. mia]⁷³¹.

Anticipò, tuttavia, inoltre che le valvole sarebbero presto state sostituite da componenti più efficienti e che la logica delle macchine sarebbe evoluta. Stimò infine che un calcolatore necessitasse di circa 10^4 organi di commutazione, previsione confermata dal fatto che questo limite venne superato solo con l’avvento di transistor e circuiti integrati [Cfr. Aspray, 1990, p.191, von Neumann, 1951, pp.13-14].

Accanto alle dimensioni e alla tipologia di materiale che li rende inaffidabili, per gli automi artificiali si presenta un terzo tipo di limitazione al raggiungimento della complessità di quelli naturali, ed è di “carattere intellettuale e non fisico” [von Neumann, 1951, p.151 trad. it.].

Questo impedimento è dovuto al fatto che “siamo ben lontani dal possedere una teoria degli automi degna di tale nome, cioè una teoria logico-matematica vera e propria”. Per affrontare matematicamente l’analisi degli automi si deve incorporare nella teoria la logica formale, ma questa sembra presentare delle carenze. “La ragione di e ciò è che essa ha a che fare con concetti rigidi, del tipo ‘tutto o niente’, ha ben pochi rapporti coi concetti continui di numero reale o di numero complesso, cioè con l’analisi matematica” [von Neumann, 1951, p.151 trad. it.]. Una teoria degli automi che li tratti come entità digitali “sembrerebbe perciò che essa debba partecipi pare di questa poco apprezzabile proprietà della logica formale, e che, dal punto di vista matematico, debba essere piuttosto combinatoria che analitica”, ma specifica von Neumann “Ora, a me pare che non sia così” [von Neumann, 1951, p.152 trad. it.]. Come procedere allora?

La logica formale è stata, secondo von Neumann, è stata applicata dai modelli di Turing e di McCulloch e Pitts agli automi, ma le mancanze di queste teorizzazioni sono dovute al fatto che con gli automi artificiali è comparsa una caratteristica inedita nella storia della logica formale. “Tutta la logica moderna la sola cosa che importi è un risultato può essere raggiunto in un numero finito di passi elementari oppure no. L’entità del numero dei passi necessari, d’altro canto, è una questione che non interessa la logica formale. Ogni successione finita di passi corretti è, in linea di principio, altrettanto buona di una qualsiasi altra” [von Neumann, 1951, p.152 trad. it.].

⁷³¹ “Thus it is the inferiority of our materials, compared with those used in nature, which prevents us from attaining the high degree of complication and the small dimensions which have been attained by natural organisms”.

Questo non è più valido per la logica dei calcolatori: “Nel caso di un automa non importa solo se esso può raggiungere un certo risultato in un numero finito di passi, ma anche quanti passi sono necessari”. Vi sono due ordini di ragioni di tipo pratico, legati ai motivi stessi per cui i calcolatori sono stati costruiti, e cioè la gestione di procedure computazionali che eccedono le capacità umane in termini quantitativi e di tempo. “La prima è che gli automi son costruiti per raggiungere certi risultati entro limiti di tempo prefissati, almeno di ordine di grandezza prefissato” e la seconda è quella per cui “i componenti adoperati hanno, in ogni singola operazione, una probabilità di errore che è piccola ma non nulla. In una catena di operazioni sufficientemente lunghe, l'ammontare complessivo di queste singole probabilità di errore può raggiungere, se incontrollato, l'ordine di grandezza dell'unità, al qual punto esso produce in effetti una assoluta mancanza di affidabilità” [von Neumann, 1951, p.152 trad. it.].

La proposta di von Neumann è quella di una modifica in due importanti aspetti della logica formale, in modo che essa possa essere applicata agli automi artificiali:

1. Dovrà tener conto della lunghezza effettiva delle catene di ragionamento, cioè delle catene di operazioni.
2. Dovrà trattare le operazioni logiche (sillogismi, congiunzioni, disgiunzioni, negazioni ecc., cioè, nella terminologia in uso per gli automi, varie forme di azioni di sbarramento, coincidenza, anticoncordanza, blocco ecc.) con procedimenti che consentano eccezioni (malfunzionamenti) con probabilità bassa ma non nulla. [von Neumann, 1951, p.153 trad. it.].

Quello che sta proponendo il matematico ungherese è un modello teorico “la cui natura sarà molto meno rigidamente a ‘tutto o niente’ che non la logica formale del passato o attuale”, più analitica che combinatoria, e con uno strettissimo legame con la *teoria probabilistica dell'informazione*, sviluppata anche grazie ai contributi di Wiener e Shannon e che si collega concettualmente coi vari tentativi di “afferrare” attraverso il calcolatore e gli strumenti artificiali la dimensione stocastica della realtà nella sua dimensione informazionale.

Afferma infatti von Neumann: “In effetti, vari elementi ci inducono a credere che questo nuovo sistema di logica formale si avvicinerà a un'altra disciplina che in passato ha avuto pochi legami con la logica. Mi riferisco alla termodinamica, soprattutto nella forma conferitale da Boltzmann, che è la parte della fisica teorica che più si avvicina in alcuni suoi aspetti alla manipolazione misurazione dell'informazione” [von Neumann, 1951, p.153 trad. it.].

La mancanza di una teoria adeguata degli automi si nota soprattutto riguardo alla questione degli *errori* e dei *malfunzionamenti* che si verificano negli organismi biologici e nei sistemi artificiali: “Gli organismi viventi verificano malfunzionamenti di componenti. L’organismo ha certo il modo di scoprirli e neutralizzarli”. Calcolate nell’ordine di 10^{20} le operazioni compiute dal sistema nervoso umano durante l’arco medio di una vita, “è un fatto che durante questa catena di eventi non si verificano mai guasti che non possano essere corretti dall’organismo stesso, senza bisogno di alcun intervento esterno determinante”. Il sistema biologico allora deve contenere in sé “dispositivi necessari per diagnosticare gli errori non appena si verificano, per regolare l’organismo in modo da minimizzare gli effetti degli errori, e infine per rettificare o bloccare permanentemente componenti difettosi”. Gli organismi viventi sono “fatti in modo da rendere gli errori quanto più piccoli e innocui è possibile” e “sono abbastanza ben congegnati da poter operare anche quando si verificano in essi dei malfunzionamenti” [von Neumann, 1951, p.154 trad. it.].

Per quanto riguarda gli automi artificiali il metodo è diverso, “si cerca in ogni modo di mettere in evidenza ogni errore non appena esso si verifica, mediante controlli matematici o automatici, dopodiché si tenta di isolare al più presto possibile il componente che ha causato l’errore [...] Una volta identificato il componente difettoso, esso viene immediatamente riparato o sostituito”. Al contrario degli automi biologici che minimizzano l’errore, in quelli artificiali, “invece, si richiede una diagnosi immediata”. Al fine di provvedere all’immediata sostituzione della componente difettosa e bloccare la propagazione dell’errore, e “gli organismi naturali sono fatti in modo da rendere gli errori quanto più piccoli e innocui è possibile; gli automi artificiali, invece, sono progettati in modo da rendere gli errori quanto più grandi e disastrosi è possibile” [von Neumann, 1951, p.154 trad. it.].

Commentando il lavoro di von Neumann riguardo al tema degli errori - tema chiave del rapporto comparato tra sistemi artificiali e naturali - Shannon afferma “Se dobbiamo considerare il cervello come una macchina, è evidente che esso è organizzato per proteggersi dagli errori in un modo del tutto diverso rispetto alle macchine di calcolo” [Shannon, 1958, p.124 trad. mia]⁷³².

La differenza tra sistemi biologici e artificiali, evidenziata in relazione ai malfunzionamenti, rende necessario fare un passo avanti - prima di tornare al Simposio Hixon - nelle considerazioni di von Neumann sugli automi, per approfondirne il legame con la teoria

⁷³² “If we are to view the brain as a machine, it is evidently organized for protection against errors in a way quite different from computing machines”.

dell'informazione. Nel dicembre del 1949 il matematico fu invitato a tenere, presso l'Università dell'Illinois, un ciclo di cinque conferenze intitolato *The Theory and Organization of Complicated Automata* [ed. Burks, von Neumann, 1966, pp.31-74].

Nella prima, von Neumann riprese i temi del Simposio Hixon sul cervello e il computer come processori di informazione, distinguendo tra dispositivi analogici e digitali, introducendo, tuttavia un tema nuovo, intorno ad un fatto non scontato: la misurazione del grado di complessità di un calcolatore. "Non è affatto ovvio come misurare la complessità di un automa" [ed. Burks, von Neumann, 1966, p. 36 trad. mia]⁷³³. Per rispondere a questo problema Von Neumann suggerì di valutare la complessità di un automa in base al numero di valvole utilizzate, alla velocità di moltiplicazione della macchina e all'efficienza, misurata in termini percentuali, delle operazioni di manipolazione del codice [Cfr. Aspray, 1990, p.201; ed. Burks, von Neumann, 1966, pp. 36-39].

Una considerazione a latere, fatta da von Neumann, durante questa lezione merita di essere sottolineata, in quanto è emblematica delle idee dello scienziato di Budapest intorno al rapporto *uomo-informazione* e alla natura della relazione di continuità tra il processore di informazioni biologico (il cervello) e quello artificiale che richiama la criticità dei *colli di bottiglia*:

Anche se non appartiene strettamente all'argomento, lasciatemi osservare che probabilmente non vorremo produrre enormi quantità di materiale numerico con le macchine calcolatrici, per esempio enormi tavole di funzioni. Il motivo per cui si utilizza una macchina di calcolo veloce non è quello di produrre una grande quantità di informazioni. Dopotutto, il solo fatto che si desideri un'informazione implica che in qualche modo si immagina di poterla assorbire e, quindi, *ovunque possano esserci colli di bottiglia nel sistema automatico* che produce ed elabora questa informazione, *esiste un collo di bottiglia ancora peggiore nell'intelletto umano* in cui, in ultima analisi, l'informazione va a riversarsi [ed. Burks, von Neumann, 1966, p. 36 trad. mia enfasi mia]⁷³⁴.

Nella seconda lezione, von Neumann, analizzò sottolineandone l'aspetto matematico, i contributi di McCulloch e Pitts e di Turing. Difese il modello neuronale di *A Logical Calculus*, che

⁷³³ "It is not completely obvious how to measure the complexity of an automaton".

⁷³⁴ "Although it doesn't belong strictly to the subject, let me point out that we will probably not want to produce vast amounts of numerical material with computing machines, for example, enormous tables of functions, The reason for using a fast computing machine is not that you want to produce a lot of information. After all, the mere fact that you want some information means that you somehow imagine that you can absorb it, and, therefore, wherever there may be bottlenecks in the automatic arrangement which produces and processes this information, there is a worse bottleneck at the human intellect into which the information ultimately seeps".

aveva avuto scarsa risonanza nella comunità biomedica, enfatizzando come questo rappresentasse l'assiomatizzazione di un "neurone idealizzato" [ed. Burks, von Neumann, 1966 p.44 trad. mia]⁷³⁵ e sottolineò il suo valore nel trattare la grande complessità degli organismi biologici visti come elaboratori dell'informazione [Cfr. ed. Burks, von Neumann, 1966 pp. 43-45]. Inoltre, "egli dimostrò la potenza dei semplici e idealizzati mattoni di McCulloch e Pitts costruendo a partire da essi alcune reti che potevano avere applicazioni utili: una memoria semplice, un meccanismo di conteggio e un circuito elementare di apprendimento" [Aspray, 1990 p.195 trad. mia]⁷³⁶.

Il modello di McCulloch e Pitts rappresentava per von Neumann un approccio "sintetico", che analizzava "strutture costruite a partire da elementi molto semplici, così che tutto ciò che occorre definire assiomaticamente sono gli elementi stessi; la loro combinazione può poi risultare estremamente complessa". A questo contrappose il lavoro di Turing, il metodo "integrale", che prendeva in esame "assiomaticamente ciò che l'intero automa deve essere, senza specificare quali siano i suoi elementi, ma soltanto definendo come esso deve funzionare" [ed. Burks, von Neumann, 1966 p.43 trad. mia]⁷³⁷. La preferenza di von Neumann sembrava essere per il metodo integrale del matematico inglese⁷³⁸.

A proposito dell'universalità della macchina di Turing, dopo aver mostrato che poteva imitare qualunque altra macchina di Turing, anche una di complessità maggiore, von Neumann osservò come "un grado minore di complessità in un automa può essere compensato da un opportuno aumento della complessità delle istruzioni". In questo, il padre dell'EDVAC vedeva il

⁷³⁵ "They wanted to axiomatize an idealized neuron".

⁷³⁶ "He demonstrated the potency of McCulloch and Pitts's simple, idealized building blocks by constructing from them some networks that could have useful applications: a simple memory, a counting mechanism, and an elementary learning circuit".

⁷³⁷ I'm going to describe both the work of McCulloch and Pitts and the work of Turing, because they reflect two very important ways to approach the subject: the synthetic way and the integral way. McCulloch and Pitts described structures built up from very simple elements, so that all you have to define axiomatically are the elements themselves, and then their combination can be extremely complex. Turing, on the other hand, started by axiomatically describing what the whole automaton is supposed to be, without specifying what its elements are, but only by describing how it is supposed to function".

⁷³⁸ Come nota William Aspray [Aspray, 1990, p.195] la preferenza di von Neumann per il modello "integrale" di Turing nella gestione di alti livelli di complessità si può desumere dal modo in cui il matematico ungherese nota che "l'intuizione che una rete neurale formale possa fare qualsiasi cosa si possa descrivere a parole è un'intuizione molto importante e semplifica enormemente le cose ai bassi livelli di complessità. Non è affatto certo che rappresenti una semplificazione ai livelli di complessità elevata. È perfettamente possibile che, ad alti livelli di complessità, il valore del teorema sia in direzione opposta, cioè che si possano esprimere logiche in termini di questi procedimenti, ma che il contrario possa non essere vero." [The insight that a formal neuron network can do anything which you can describe in words is a very important insight and simplifies matters enormously at low complication levels. It is by no means certain that it is a simplification on high complication levels. It is perfectly possible that on high complication levels the value of the theorem is in the reverse direction, namely, that you can express logics in terms of these efforts and the converse may not be true, ed. Burks, von Neumann, 1966, pp.47-48].

grande merito di Turing: “Se si costruisce correttamente un automa, allora qualsiasi ulteriore requisito riguardante l’automa può essere gestito tramite istruzioni sufficientemente elaborate” [ed. Burks, von Neumann, 1966, p. 50, trad. mia]⁷³⁹.

La terza lezione ci porta al cuore del tema della teoria dell’informazione e dell’idea, con cui avevamo lasciato l’esposizione al Simposio Hixon, di “estendere la teoria logica degli automi a una teoria probabilistica” [Aspray, 1990, p. 198, trad. mia]⁷⁴⁰. Il problema affrontato da von Neumann in questo contesto è quello dell’affidabilità dei materiali. Come ha mostrato l’analisi comparata degli errori e dei guasti nei sistemi biologici e artificiali, tale questione costituiva la ragione principale per estendere la logica formale a una logica probabilistica, così da massimizzare l’affidabilità degli automi.

“Tutto ciò che possiamo cercare di fare è disporre un automa in modo che, nella stragrande maggioranza dei guasti, possa continuare a funzionare. Queste disposizioni forniscono palliativi ai guasti, non cure. La maggior parte delle disposizioni degli automi artificiali e naturali e i principi su cui si basano sono di questo tipo” [ed. Burks, von Neumann, 1966, p. 50, trad. mia]⁷⁴¹.

Von Neumann illustra alcuni sviluppi della teoria probabilistica che potevano essere integrati nella nuova logica degli automi, sostituendo l’impostazione deterministica con una logica in cui gli assiomi non conducono a un unico esito, ma definiscono piuttosto una distribuzione di probabilità sui risultati possibili. “Consentire il guasto come entità logica indipendente significa non enunciare gli assiomi in maniera rigorosa. Gli assiomi non sono della forma: se A e B accadono, allora seguirà C. Gli assiomi sono sempre di questo tipo: se A e B accadono, allora C seguirà con una certa probabilità specificata” [ed. Burks, von Neumann, 1966, p. 58, trad. mia]⁷⁴².

La distinzione - già chiara, secondo von Neumann, a Laplace, e successivamente ripresa e riformulata da Keynes (1883-1946) con il suo approccio logico-relazionista [Keynes, 1921] -

⁷³⁹ “Thus a lesser degree of complexity in an automaton can be compensated for by an appropriate increase of complexity of the instructions. The importance of Turing’s research is just this: that if you construct an automaton right, then any additional requirements about the automaton can be handled by sufficiently elaborate instructions

⁷⁴⁰ In the third Illinois lecture, “Statistical Theories of Information,” von Neumann described his ideas for extending the logical theory of automata to a probabilistic theory”.

⁷⁴¹ “All we can try to do is to arrange an automaton so that in the vast majority of fallures it can continue to operate. These arrangements give palliatives of failures, not cures. Most of the arrangements of artificial and natural automata and the principles involved therein are of this sort”.

⁷⁴² “To permit failure as an independent logical entity means that onedoes not state the axioms in a rigorous manner. The axioms are notof the form: if A and B happen, C will follow. The axioms are always ofthis variety: if A and B happen, C will follow with a certain specifiedprobability, D will follow with another specified probability, and soon”.

viene intesa come misura del grado di implicazione o di conferma di una proposizione rispetto a un'altra, estendendo in tal modo l'ambito della logica classica [Cfr. ed. Burks, von Neumann, 1966, p. 59].

Pur con le cautele del caso, inoltre, von Neumann osserva come anche la meccanica quantistica orienta fortemente verso una concezione in cui "la probabilità è intrinsecamente legata alla logica" [ed. Burks, von Neumann, 1966, p. 59 trad. mia]⁷⁴³.

Nella terza lezione all'Università dell'Illinois, tuttavia, von Neumann non presentò una teoria formale della probabilità. Queste considerazioni, però, permettono di mettere in relazione l'estensione della logica formale - legata al tema dell'affidabilità e degli errori negli automi - con due filoni di ricerca: "Il lavoro dei termodinamici su entropia e informazione e il lavoro degli ingegneri elettrici sulla teoria della comunicazione" [Aspray, 1990, p. 198, trad. mia]⁷⁴⁴. Nello specifico, il matematico ungherese prese in considerazione, da un lato, "la teoria dell'entropia e dell'informazione in termodinamica" e, dall'altro, strettamente connessa a quest'ultima, "la teoria dell'informazione di Shannon" [ed. Burks, von Neumann, 1966, p. 59, trad. mia]⁷⁴⁵.

Il richiamo alla matematizzazione della comunicazione di Shannon e al concetto di entropia - del quale von Neumann sottolinea l'origine negli studi di termodinamica e di fisica statistica e gli sviluppi nei lavori di Nyquist e Hartley - permette di dare una spiegazione in termini informazionali dell'errore. "Paragonò i *colli di bottiglia* nei sistemi di calcolo - in particolare le memorie di piccole dimensioni e le apparecchiature di input lente - ai motori termici con eccessivi differenziali di temperatura, che li rendono inefficienti nel funzionamento" [Aspray, 1990, p.199 trad. mia]⁷⁴⁶.

Le idee di von Neumann riguardo alla metamorfosi probabilistica della teoria dell'informazione, al suo nesso con gli automi e alla centralità del concetto di entropia nel loro funzionamento sono nette:

⁷⁴³ "There's a great deal in other modern theories, for instance, in quantum mechanics, which inclines one very strongly to take this philosophical position, although the last word about this subject has certainly not been said and is not going to be said for a long time. Anyway, one is also tempted in the case of quantum mechanics to modify one's outlook on logics and to view probability as intrinsically tied to logics".

⁷⁴⁴ "These are the work of thermodynamicists on entropy and information and the work of the electrical engineers on communication theory".

⁷⁴⁵ "The first is the theory of entropy and information in thermodynamics; the second is Shannon's information theory".

⁷⁴⁶ "He compared bottlenecks in computing systems-notably small memories and slow input equipment-to heat engines that have excessive temperature differentials that make them run inefficiently".

Concetti termodinamici entreranno probabilmente in questa nuova teoria dell'informazione. *Ci sono forti indizi che l'informazione sia simile all'entropia e che i processi degenerativi dell'entropia trovino un parallelo nei processi degenerativi dell'elaborazione dell'informazione.* È probabile che non si possa definire la funzione di un automa, o la sua efficienza, senza caratterizzare l'ambiente in cui opera mediante tratti statistici simili a quelli usati per caratterizzare un ambiente in termodinamica. Le variabili statistiche dell'ambiente dell'automa saranno, naturalmente, più complesse della variabile termodinamica standard della temperatura, ma probabilmente simili nella natura [ed. Burks, von Neumann, 1966, p. 59, trad. mia enfasi mia]⁷⁴⁷.

Riguardo a queste ultime affermazioni di von Neumann, occorre sottolineare come il ruolo degenerativo dell'entropia nei processi di elaborazione dell'informazione, e nei meccanismi omeostatici che caratterizzano ogni automa, sia destinato a collocarsi coerentemente con una riflessione che, in continuità con la cibernetica wieneriana, che considera ogni entità, artificiale o naturale, come processore dinamico dell'informazione, vedrà nella rimozione dell'entropia la base della *Information Ethics* delineata da Luciano Floridi. In questa prospettiva, la perdita, l'impoverimento o la degenerazione delle strutture informative - intesa in senso non termodinamico ma ontologico, come danno agli oggetti informativi dell'ambiente - diventano un "male" da prevenire, minimizzare e, ove possibile, eliminare. [Cfr. Ward Bynum, 2009; Floridi, 2009, p. 41]⁷⁴⁸.

⁷⁴⁷ "Thermodynamical concepts will probably enter into this new theory of information. There are strong indications that information is similar to entropy and that degenerative processes of entropy are paralleled by degenerative processes in the processing of information. It is likely that you cannot define the function of an automaton, or its efficiency, without characterizing the milieu in which it works by means of statistical traits like the ones used to characterize a milieu in thermodynamics. The statistical variables of the automaton's milieu will, of course, be somewhat more involved than the standard thermodynamical variable of temperature, but they will probably be similar in character".

"Wiener considerava l'entropia il più grande male naturale" [Bynum, 2009, p.8]. Come visto in precedenza, "ogni ente dell'infosfera può essere danneggiato o distrutto, alterando la sua caratteristica struttura di dati, in tal modo impedendogli di svilupparsi". Tale danno o distruzione è chiamato da Floridi "entropia" [Bynum, 2009, p.17]. *L'Information Ethics*, o Etica Informazionale, innalza l'informazione - intesa come livello di astrazione comune per la considerazione di ogni ente (che a un livello ontologico minimale si configura come non contraddittorio e denotabile) - a vero soggetto paziente dell'etica. Ciò consente di superare i bias e le impasse antropocentriche delle etiche standard (si pensi, ad esempio, alle difficoltà di universalizzazione degli imperativi kantiani), così come i limiti biocentrici delle etiche non-standard, come quelle ambientaliste [Cfr. Floridi, 2009, pp.25-57; Floridi, 2008, pp.3-34]. L'Etica dell'Informazione, così configurata, si fonda su quattro regole/principi fondamentali:

0. Non si deve generare entropia nell'infosfera (legge negativa).
1. Si deve prevenire lo sviluppo dell'entropia nell'infosfera.
2. Si deve rimuovere l'entropia nell'infosfera.
3. Lo sviluppo degli enti informativi, come dell'intera infosfera, deve essere promosso, preservando, coltivando ed arricchendo le loro proprietà [Floridi, 2009, p.41].

Questo approccio permette di sfuggire sia alle strette maglie di un'interpretazione "conservativa" (quella di D. Johnson, 1985), secondo cui i problemi posti dal digitale sono risolvibili con gli strumenti dell'etica tradizionale, sia ai rischi di isolamento dal dibattito e dai metodi etici mainstream che derivano da un approccio radicale

Non è da sottovalutare neanche un aspetto “socio-tecnico” della riflessione di von Neumann che riflette il suo status di studioso e contemporaneamente di persona inserita nella rete ricerca scientifica governativa, militare e privata.

Come osserva Goldstine:

Il lavoro di von Neumann sugli automi costituiti da parti inaffidabili derivava in parte dal suo interesse per il problema, posto dall'Aeronautica, dell'affidabilità dei missili, e in parte dalle sue osservazioni su come il sistema nervoso umano ci permetta di funzionare in modo abbastanza affidabile anche se i singoli neuroni possono essere inaffidabili. Dal 1951 egli fece parte del Comitato Consultivo Scientifico dell'Aeronautica e rimase colpito dall'esigenza della massima affidabilità nel funzionamento dei missili, i quali però avevano una durata di vita di soli pochi minuti [Goldstine, 1972, p.279 trad. mia]⁷⁴⁹.

L'integrazione nella teoria dell'informazione e della dimensione probabilistica su funzionato ed errori dei componenti di automi da parte di von Neumann proseguì nel gennaio 1952. Il matematico di Budapest sviluppò ulteriormente il suo pensiero sulla costruzione di sistemi affidabili a partire da componenti inaffidabili in un ciclo di cinque lezioni al California Institute of Technology, intitolato *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components* [von Neumann, 1956]. Anche in questo luogo von Neumann considera l'errore non come un mero accidente, “ma come parte essenziale del processo in esame - la cui importanza nella sintesi degli automi è pienamente paragonabile a quella del fattore normalmente preso in considerazione, ossia la struttura logica prevista e corretta”. Ribadisce, inoltre, “che l'errore dovrebbe essere trattato con metodi termodinamici, e diventare oggetto di una teoria termodinamica, così come è avvenuto per l'informazione grazie al lavoro di L. Szilard e C. E. Shannon” [von Neumann, 1952, p.329 trad. mia]⁷⁵⁰.

all'etica dell'informazione come disciplina totalmente nuova, portatrice di problemi radicalmente inediti e bisognosa di soluzioni del tutto ad hoc [Cfr. Floridi, 2009b, pp.73-76].

⁷⁴⁹ “Von Neumann's work on automata formed out of unreliable parts was an outgrowth, in part, of his interest in the Air Force's problem of reliability of its missiles, and in part of his observations on how the the human nervous system allows us to work quite reliably even though individual neurons may be unreliable. He was on the Scientific Advisory Board of the Air Force from 1951 on and was struck with need for highest reliability of functioning of missiles which had however lives of only a few minutes”.

⁷⁵⁰ “Error is viewed, therefore, not as an extraneous and misdirected or misdirecting accident, but as an essential part of the process under consideration-its importance in the synthesis of automata being fully comparable to that of the factor which is normally considered, [...] that error should be treated by thermodynamical methods, and be the subject of a thermodynamical theory, as information has been, by the work of L. Szilard and C. E. Shannon the intended and correct logical structure”.

A differenza dei precedenti interventi in Illinois e al Simposio Hixon, tuttavia, qui von Neumann non si limitò a un'analisi descrittiva, ma costruì anche l'apparato matematico necessario per affrontare il problema in modo formale. Sulla scia di McCulloch e Pitts, "presentò gli automi come reti di neuroni idealizzati, collegati da linee che trasportano impulsi, in cui l'uscita di un neurone diventa l'ingresso di un altro. collegati da linee che trasportano impulsi: l'uscita di un neurone diventa l'ingresso di un altro" [Aspray, 1990, p.199 trad. mia]⁷⁵¹.

In essi ogni linea di ingresso può essere eccitatoria o inibitoria. A ciascun neurone è associata una funzione di soglia $\varphi(x)$, che stabilisce le condizioni per l'attivazione: un neurone si eccita al tempo $t + 1$ se, al tempo t , il numero di ingressi eccitatori k e quello di ingressi inibitori j soddisfano la relazione: $k \geq \varphi(j)$ [Cfr. Goldstine, 1972, p.280, von Neumann, 1952, p.332].

A partire da questi von Neumann dimostrò una corrispondenza biunivoca tra la classe di automi a singola output e i polinomi del calcolo proposizionale. Formulò quindi un teorema di riduzione, secondo cui qualsiasi automa a singola uscita può essere costruito combinando tre operazioni logiche fondamentali: "a OR b", "a AND b" e "a AND NOT b" [Cfr. Aspray, 1990, pp.199-200].

A partire da questo mostrò che ogni rete [*network*] può essere costruito a partire da un singolo tipo di automa, grazie all'introduzione di due organi: "lo *Sheffer Stroke* e l'organo di maggioranza" [Goldstine, 1972, p.281 trad. mia; von Neumann, 1956, pp.337-339]⁷⁵².

Il primo⁷⁵³ realizza la funzione logica $NAND(a, b) = \neg(a \wedge b)$, vera in tutti i casi tranne quando sia a che b sono veri. Con sole operazioni NAND si possono ottenere nel neurone

⁷⁵¹ "He presented the automata as networks of idealized neurons, linked by lines that carry pulses, in which the output of one neuron becomes the input of another neuron".

⁷⁵² He then introduced two organs: the well-known "Sheffer stroke" and another one he called a majority organ.

⁷⁵³ Il "colpo di Sheffer" (Sheffer stroke, in notazione "|") è un connettivo logico fondamentale, introdotto formalmente da Henry Maurice Sheffer (1882-1964) nel 1913, nella memoria *A set of five independent postulates for Boolean algebras, with application to logical constants* [Sheffer, 1913], pubblicata sui *Transactions of the American Mathematical Society*. Sheffer dimostrò che l'operatore da lui proposto (interpretato nel testo come *nondisjunction*, cioè NOR) consentiva di assiomatizzare l'algebra di Boole, e che, grazie all'autodualità - le leggi restano valide scambiando AND con OR e 0 con 1 - di quest'ultima, l'operatore risultava equivalente anche alla non-congiunzione (NAND). Fu Jean Nicod (1893-1924) nel 1917, nell'articolo *A reduction in the number of primitive propositions of logic* [Nicod, 1917], a introdurre l'uso del segno di Sheffer come simbolo della non-congiunzione (NAND), convenzione divenuta poi quella prevalente. Nella seconda edizione (1927) dei *Principia Mathematica*, Bertrand Russell e Alfred North Whitehead adottarono il simbolo di Sheffer, proponendolo come sostituto degli operatori "OR" e "NOT" della prima edizione. Tuttavia oltre trent'anni prima, C. S. Peirce (1839-1914) aveva scoperto la completezza funzionale di NAND e NOR, coniando il termine *ampheck* ("tagliare in entrambe le direzioni") e discutendone nel manoscritto inedito *The Algebra of Logic* (Peirce, 1885), ma senza mai pubblicare i suoi risultati. Infine, Edward Stamm (1886-1940 ca.) aveva descritto nel 1911 i connettivi NAND e NOR, mostrando come da essi si potessero derivare tutte le altre operazioni booleane [Stamm, 1911], anticipando di poco la formulazione di Sheffer.

idealizzato le operazioni booleane fondamentali di negazione [AND], congiunzione [OR] e disgiunzione [NOT], costituendo così una base funzionalmente completa [von Neumann, 1956, pp.338-339]⁷⁵⁴. Il secondo invece, “l’organo di maggioranza” era un “dispositivo per stabilire se almeno due dei tre ingressi coincidono”. Veniva impiegato, da qui il nome, in “molti punti del suo automa per determinare ciò che aveva deciso la maggioranza” [Goldstine, 1972, p.283 trad. mia]⁷⁵⁵. Questo secondo organo, i termini logici, calcolava $m(a, b, c) = (a \wedge b) \vee (a \wedge c) \vee (b \wedge c)$, restituendo *vero* se almeno due dei tre ingressi sono veri. Con l’aggiunta della negazione (o di ingressi costanti 0/1) anche questo organo forma una base funzionalmente completa, ossia in grado di realizzare qualunque funzione booleana.

A partire da questi due organi, e dalle reti che realizzavano von Neumann, mostrò come sintetizzare alcuni automi semplici: *un organo di memoria* [*memory unit*] capace di ricordare se l’ultimo ingresso stimolato sia stato *a* o *b*: uno *scaler* per contare gli stimoli d’ingresso a due a due e, a partire da quest’ultimo, un automa semplice che apprende [*learning device*], che impara ad anticipare lo stimolo *b* quando questo segue regolarmente lo stimolo *a* [Cfr. von Neumann, 1952, pp.342-345].

Questi automi, se dotati di cicli di retroazione - la cui assenza costituisce il limite principale del modello di McCulloch e Pitts - ampliano la logica delle macchine costruttive a gran parte della logica intuizionista, ossia a quell’insieme di enunciati per i quali esiste una costruzione o procedura effettiva che ne dimostri la verità; con l’aggiunta di una memoria illimitata, come un nastro infinito corredato di organi di lettura e scrittura, acquisiscono secondo von Neumann un potere computazionale pienamente equivalente alla logica intuizionista e a una macchina di Turing universale [Cfr. von Neumann, 1952, pp. 340-341].

Tali sistemi furono poi connessi al problema degli errori, nella prospettiva di integrare la teoria della probabilità e dell’informazione all’interno di una teoria degli automi. “In tutte le considerazioni precedenti si è assunto che i componenti di base fossero impeccabili nel loro funzionamento. Tale assunzione non è chiaramente molto realistica. I dispositivi meccanici, così come quelli elettrici, sono statisticamente soggetti a guasti, e lo stesso è probabilmente vero

⁷⁵⁴ Per la rappresentazione grafica, nei modelli di automa neurone/circuito e degli operatori logici - come lo *Sheffer stroke* o l’organo di maggioranza, impiegati per la loro realizzazione - analizzati da von Neumann in *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*, si veda von Neumann 1952, pp. 332-340 e Goldstine 1972, pp. 280-283.

⁷⁵⁵ “Von Neumann’s majority organ was a device for deciding whether at least two out of three inputs agree. He was to use it at many places in his automaton to decide what the majority had decided”.

anche per i neuroni animali” [von Neumann, 1952, p.345 trad. mia]⁷⁵⁶. Un modo “realistico” di affrontare il tema dell’errore è quello di considerare “i malfunzionamenti [...] statisticamente dipendenti dallo stato generale della rete e tra loro” [von Neumann, 1952, p.345 trad. mia]⁷⁵⁷.

Il rischio dell’inaffidabilità dei componenti non era per von Neumann tanto quello legato al singolo errore sporadico, “quanto il degrado progressivo che mette in dubbio l’affidabilità complessiva” [Aspray, 1990, p.200 trad. mia]⁷⁵⁸.

Il matematico fornì un esempio servendosi dell’automa di memoria appena formalizzato. Quando questo viene realizzato con componenti soggetti a malfunzionamento, la probabilità che svolga correttamente la propria funzione si riduce, col passare del tempo, fino a stabilizzarsi intorno a $1/2$, annullando di fatto la sua capacità di memoria. La formulazione data da von Neumann al problema fu esposta in questi termini: “data una qualunque probabilità di errore fissa $e < 1/2$ nei componenti di base (organi di Sheffer o organi di maggioranza) e un valore $\delta > 0$, è possibile costruire, a partire da questi componenti di base, un automa che esegua la funzione desiderata e che commetta un errore nell’output con probabilità inferiore a δ ” [Aspray, 1990, p.200 trad. mia]⁷⁵⁹

La proposta del matematico ungherese fu quella di “progettare il suo automa in modo tale che, anche se ciascun neurone avesse una probabilità finita di errore, la risposta degli output finali risultasse comunque arbitrariamente affidabile” [Goldstine, 1972, p.283 trad. mia]⁷⁶⁰. La risposta alla domanda precedente intorno alla possibilità di tenere la probabilità di errore sotto un valore prefissato, implicava, integrando i concetti della teoria dell’informazione di Shannon, l’impiego della “ridondanza”. Qui possiamo notare una torsione, almeno teorica, del *design* degli automi verso una struttura più simile a quella del processore biologico di informazioni. “In questo senso, esse rispecchiano l’architettura del sistema nervoso centrale.”

⁷⁵⁶ “In all the previous considerations, it has been assumed that the basic components were faultless in their performance. This assumption is clearly not a very realistic one. Mechanical devices as well as electrical ones are statistically subject to failure, and the same is probably true for animal neurons too”.

⁷⁵⁷ “A more general assumption, which is on a good deal more realistic, is this: The malfunctions are statistically dependent the general state of the network and on each other”.

⁷⁵⁸ “The problem with unreliable components, he claimed, is not that they will occasionally cause the automaton to give an incorrect result but rather that they degrade the performance such that the reliability of all results is called into question”.

⁷⁵⁹ “With the problem carefully specified, he set his problem mathematically as follows: given any fixed probability of error $e < 1/2$ in the basic components (Sheffer organs or majority organs) and $\delta > 0$, can an automaton be built from these basic components that will perform the desired function and commit an error in the output of the automaton with probability less than δ ?”

⁷⁶⁰ “Yon Neumann was concerned with the problem of how to design his automaton so that even if each neuron in it had a finite probability of error, nonetheless the response of the final outputs would be arbitrarily reliable”.

[Goldstine, 1972, p.283 trad. mia]⁷⁶¹. La soluzione proposta da von Neumann in questa occasione fu quella del *multiplexing*, l'impiego, per ridurre l'errore a piacere, di fasci [*bundles*] di linee, invece che linee singole⁷⁶².

Tuttavia, von Neumann era ben consapevole dei limiti delle tecniche *multiplexing* individuate. Riconosceva che risultavano, a livello implementativo, "impraticabili con le tecnologie dei componenti attuali (sebbene questo forse non sia vero per certe tecnologie concepibili in futuro), ma non sono necessariamente irragionevoli (almeno non per motivi di dimensioni) per la micro-componentistica del sistema nervoso umano" [von Neumann, 1952, p.368 trad. mia]⁷⁶³.

Proprio in relazione alla componentistica fisiologica, è possibile individuare un altro limite della soluzione proposta. Nell'analogia tra automi biologici e artificiali, una delle ragioni principali dei limiti del modello era che tale impostazione "seguiva troppo pedissequamente un piano costruttivo fisso - e in particolare uno ispirato alle procedure convenzionali della matematica e della logica matematica" [von Neumann, 1952, p.368 trad.mia]⁷⁶⁴.

Von Neumann ammetteva, certo, che i "sistemi nervosi animali obbediscono a certi schemi 'architettionici' rigidi nella loro costruzione su larga scala", ma precisava anche che, a un livello di analisi più granulare "nei dettagli più fini e a livello microscopico" - qui si coglie la piena coerenza con il programma di ricerca già delineato nella lettera a Wiener del novembre 1946 - questi si prestavano ad un'analisi statistica più flessibile. [von Neumann, 1952, p.369 trad.mia]⁷⁶⁵.

⁷⁶¹ "His solutions require the use of redundancy. In this sense they mirror the central nervous system's design".

⁷⁶² Questo "trick" [Goldstine, 1972, p.284], consisteva nell'invio dello stesso segnale su un fascio di N linee invece che su una sola, interpretando come "vero" la stimolazione di almeno $(1 - \theta) \cdot N$ linee, come "falso" quella di meno di $\theta \cdot N$ linee, e come "indeciso" il caso intermedio, con $0 < \theta < \frac{1}{2}$. Ogni neurone è arricchito da un organo esecutivo [*basic executive organ*], che elabora i segnali in ingresso, e da un organo di ripristino [*restoring organ*], che riporta la stimolazione a un livello corretto [von Neumann, 1952, pp.353-356]. Un'analisi statistica mostra che aumentando N si può ridurre l'errore a piacere; stime di von Neumann indicavano che sarebbero servite circa 20.000 linee ($\pm 40\%$) per ottenere l'affidabilità richiesta. [Cfr. Aspray, 1990, p.201]. L'analisi statistica formalizzata da von Neumann mostrava come si poteva ridurre l'errore a piacere; stime di von Neumann indicavano che sarebbero servite circa 20.000 linee ($\pm 40\%$) per ottenere l'affidabilità richiesta [Cfr. von Neumann, 1952, p.367-369].

⁷⁶³ "This implies, such techniques are impractical for present technologies of componentry (although this may perhaps not be true for certain conceivable technologies of the future), but they are not necessarily unreasonable (at least not on grounds of size alone) for the micro-componentry of the human nervous system".

⁷⁶⁴ "The main trouble with the multiplexing systems, as of described in the preceding section, is that they follow too slavishly a fixed plan too, construction-and specifically one, that is inspired by the conventional procedures of mathematics and mathematical logics".

⁷⁶⁵ "It is true, that the animal nervous systems, obey some rigid "architectural" patterns in their large-scale construction, and that those variations, which make one suspect a merely statistical design, seem to occur only in finer detail and on the micro-level".

Come abbiamo già osservato analizzando le origini della cibernetica wieneriana, il trattamento di un sistema come *black box* comporta una modellizzazione di tipo probabilistico, capace di attenuare le rigidità della logica formale tradizionalmente impiegata nei modelli di automi e, al tempo stesso, di dar conto dei possibili malfunzionamenti. Questo punto è riconosciuto dallo stesso von Neumann: le “scatole nere” a permutazione “casuale” [...] sono tipiche di un “disegno meramente statistico” [von Neumann, 1952, p.369 trad.mia]⁷⁶⁶.

Tuttavia von Neumann introduce un’importante distinzione. Nonostante il guadagno in termini di adattabilità nel design di un automa in termini di teoria dell’informazione, il “sistema nervoso sembra essere progettato in modo più flessibile”. Questa differenza, che analizzeremo meglio a breve, è individuata nel fatto che “le sue operazioni “digitali” (neuronal) si alternano piuttosto liberamente con i processi “analogici” (umoral) nelle loro catene causali complete” [von Neumann, 1952, p.369 trad.mia]⁷⁶⁷.

Se la metafora cibernetica impiegata per il calcolatore a programma memorizzato, come EDVAC, postulava un’architettura seriale per calcolatore e cervello, ora von Neumann riconosce che:

L’intero schema logico del sistema nervoso sembra deviare in alcuni tratti importanti, qualitativamente e in modo significativo, dai nostri comuni modelli matematici e logico-matematici: i treni di impulsi che trasportano messaggi “quantitativi” lungo le fibre nervose non sembrano essere espressioni digitali codificate (come una digitalizzazione binaria o decimale codificata in [Morse o binario]) di un numero, ma piuttosto espressioni “analogiche” di esso [von Neumann, 1952, p.369 trad.mia]⁷⁶⁸.

“Pertanto”, conclude von Neumann, “è probabile che esistano profonde differenze nei principi organizzativi di base” [von Neumann, 1952, p.369 trad.mia]⁷⁶⁹.

Approfondiremo a breve l’analisi comparata tra cervello e calcolatore, nella quale culmina la riflessione di von Neumann sugli automi. È tuttavia importante rilevare come la

⁷⁶⁶ “It is true, that our multiplexing scheme, too, is rigid only in its large-scale pattern ...] while the “random” permutation “black boxes” [...] are typical of a “merely statistical design”.”.

⁷⁶⁷ “Yet the nervous system seems to be somewhat more flexibly designed. Also, its “digital” (neural) operations are rather freely alternating with “analog” (humoral) processes in their complete chains of causation.”.

⁷⁶⁸ “Finally the whole logical pattern of the nervous system seems to deviate in certain important traits qualitatively and significantly from our ordinary mathematical and mathematical-logical modes of operation: The pulse-trains that carry “quantitative” messages along the nerve fibres do not seem to be coded digital expressions (like a binary or a [Morse or binary coded] decimal digitalization) of a number, but rather “analog” expressions of one”.

⁷⁶⁹ “Thus deep differences in the basic organizational principles are probably present”.

ricomprensione della teoria probabilistica - che lo aveva condotto a riformulare la logica per l'analisi degli automi naturali e artificiali come processori di informazione in grado di gestire errori, malfunzionamenti e colli di bottiglia - abbia messo in evidenza alcuni aspetti fondamentali: il carattere solo parzialmente digitale del neurone; la necessità di studiarne il comportamento all'interno di reti; e la maggiore plasticità di queste ultime nel fronteggiare l'errore. Entrare nella struttura profonda delle componenti degli automi, mostra *in nuce* un mutamento degli assunti delle strutture computazionali.

L'*excursus* appena fatto e l'aver notato la presenza di differenze architetture nei tipi emerse dalle analisi dell'affidabilità dei componenti degli automi (e la necessità di riformarne la logica) ci permette di tornare alle considerazioni di von Neumann intorno a questi temi esposte al simposio Hixon del 1948 in modo da valutarne conseguenze importanti.

L'analisi dei modelli computazionali degli automi aveva messo in luce la necessità di ampliarne la logica in senso probabilistico. Il risultato di McCulloch e Pitts era secondo von Neumann "il risultato più significativo fin qui ottenuto col metodo assiomatico" [von Neumann, 1951, p.157 trad. it.]. Il loro guadagno maggiore era stato il mostrare

Che tutto ciò che può essere descritto esaurientemente e senza ambiguità, tutto ciò che può essere tradotto in parole completamente e senza ambiguità, può, per ciò stesso, essere realizzato mediante una appropriata rete nervosa finita. Poiché è ovvio l'inverso, si può perciò dire che non v'è differenza tra la possibilità di descrivere a parole completamente e senza ambiguità un tipo di comportamento reale o immaginario e la possibilità di realizzarlo mediante una rete formale nervosa [von Neumann, 1951, p.157 trad. it.].

Tuttavia si presentano due ordini di difficoltà. La prima, di carattere pratico - già riconosciuta da McCulloch - riguarda il fatto che, anche ammesso che "un certo tipo di comportamento può essere realizzato mediante una rete formale nervosa, rimane aperta ancora la questione se quella rete possa essere realizzata entro dimensioni pratiche, e precisamente se essa si adatterà alle limitazioni fisiche dell'organismo in questione". La seconda difficoltà, invece, è di ordine logico: "se ogni tipo di comportamento esistente possa essere realmente tradotto in parole completamente e senza ambiguità" [von Neumann, 1951, p. 158, trad. it.].

Riguardo a questo secondo punto, von Neumann osserva che, se ogni fase elementare di un comportamento può essere descritta "completamente e senza ambiguità", rimane aperto il

problema di estendere ciò al comportamento complessivo. Ne è un esempio l'analogia visiva. Un organismo può riconoscere come triangoli figure regolari, incomplete, curvilinee o tratteggiate, ma tentare di enumerare tutte le varianti produrrebbe descrizioni smisurate e inevitabilmente indefinite, dando la "vaga e sconcertante sensazione" di un compito senza fine che richiederebbe la descrizione in "migliaia o milioni o un numero altrettanto impossibile di volumi". [Cfr, von Neumann 1951, p.159 trad. it.]. Probabilmente, secondo von Neumann, "il modo più semplice, e il solo pratico attualmente, per dire che cosa sia un'analogia visiva consista nel dare una descrizione delle connessioni dell'area visiva del cervello" [von Neumann 1951, p.159 trad. it.]. Ma in questo caso, prosegue il matematico ungherese, "ci imbattiamo qui in parti della logica di cui non si ha praticamente alcuna esperienza precedente. L'ordine di complessità è tale da non poter essere confrontato con nulla di ciò che si è potuto conoscere" "A questo livello il risultato di McCulloch-Pitts non serve più" [von Neumann, 1951, p.160 trad. it.].

C'è un limite nella formalizzazione assiomatica delle reti neurali. Ciò che von Neumann ritiene necessario, a questo livello di complessità, è un'ulteriore modifica della teoria logica. Quest'ultima deve assumere in sé i tratti dello studio neurofisiologico – a questo fine si sono messe in luce via via le differenze tra l'automa biologico ad alta complessità e quello artificiali - del cervello:

Tutto ciò non altera la mia opinione che sia necessaria una nuova teoria essenzialmente logica, per comprendere gli automi ad alta complessità e, in particolare, il sistema nervoso centrale. Può darsi tuttavia che in questo processo la logica debba subire una pseudomorfosi, trasformandosi in neurologia in misura molto maggiore che non l'inverso. L'analisi precedente mostra che una delle cose importanti che si possono fare al presente, riguardo la teoria del sistema nervoso centrale, è di indicare dove non sta il vero problema [von Neumann, 1951, p.160 trad. it.].

Questo terzo *step* di riformulazione della logica, che segue i due stadi precedenti che avevano riguardato il calcolo delle catene di operazioni e l'introduzione della probabilità per trattare errori e malfunzionamenti, apre lo spazio ad una domanda. Come trattare ordini di complessità per cui persino l'assiomatizzazione così ampliata non vale più?

La complessità, secondo von Neumann, non è solo di ordine quantitativo, ma ha anche un aspetto qualitativo, che può essere mostrata attraverso quello che sembra un apparente "circolo vizioso" degli automi biologici", "la più semplice espressione del quale è che gli

organismi molto complicati possono riprodursi.” [von Neumann, 1951, p.160 trad. it.]. Qui von Neumann introduce un ulteriore concetto fondamentale, oltre a quello della complessità, nell’analisi degli automi: quello di auto-replicazione.

Questo concetto può apparire a tutta prima paradossale. “Ci si aspetta che la complessità di una specie aumenti (o almeno rimanga costante) attraverso l’evoluzione, e tuttavia è difficile comprendere come un automa possa produrre qualcosa di più complesso di sé stesso” [Aspray, 1990, p.193 trad. mia]⁷⁷⁰. Intuitivamente, se un automa è capace di costruirne un altro, “deve aversi una diminuzione di complicazione quando si passa dal genitore al costrutto. Cioè, se A può produrre B, B. allora A deve contenere in qualche modo una descrizione completa” [von Neumann, 1951, p.160 trad. it.]. Inoltre, A deve possedere gli strumenti per costruire l’automa B. Da qui si potrebbe dedurre come la degenerazione e la diminuzione della complessità siano la tendenza generale. Tuttavia “ciò è in evidente contraddizione con i fatti più evidenti che hanno luogo in natura. Gli organismi si riproducono, cioè producono nuovi organismi, senza diminuire di complessità. Per giunta vi sono lunghi periodi di evoluzione, durante i quali la complessità è addirittura crescente” [von Neumann, 1951, p.160 trad. it.].

Per dar conto di questo fenomeno, von Neumann prese come riferimento la macchina universale di Turing, la quale è “è in grado di elaborare qualsiasi successione prodotta da una qualsiasi macchina”. In questo, secondo von Neumann risiede il guadagno maggiore del lavoro di Turing: “Come può esistere una macchina che compia lo stesso lavoro di una qualsiasi altra possibile macchina, per esempio di una macchina che abbia il doppio delle sue dimensioni e della sua complessità?” “Si potrebbe credere a priori che ciò sia impossibile [...] Turing tuttavia provò che ciò è possibile” [von Neumann, 1951, p.160 trad. it.]. Malgrado questo guadagno fondamentale, von Neumann, in linea con la lettera a Wiener analizzata in precedenza, considera questo risultato “troppo ristretto solo in un punto. I suoi automi sono solo macchine calcolatrici. La loro uscita è un pezzo di nastro che porta impressi degli zero e degli uno. Ciò che occorre per la costruzione cui mi riferivo è un automa la cui uscita sia costituita da altri automi”. Ma è possibile ottenere un “equivalente del risultato di Turing”, grazie al quale trattare logicamente e matematicamente le componenti base, elementari, degli automi [von Neumann, 1951, p.161 trad. it.].

⁷⁷⁰ “He formulated his analysis in terms of a paradox involving self-replication: one expects the complexity of a species to increase (or at least remain constant) through evolution, yet it is hard to understand how an automaton can produce anything more complicated than itself”.

A tal scopo, alla conferenza Hixon, von Neumann presentò un modello cinematico [*kinematic*] di automa. A partire da un set ben definito, arbitrariamente lungo, di “parti elementari della macchina” dotato di “definizione operativa” - in continuità con quanto già affermato nella lettera a Wiener del 29 novembre 1948 - si può mostrare che è sufficiente una dozzina circa di parti elementari. Il problema dell'autoriproduzione, viene riformulato da von Neumann in questi termini, cioè se sia possibile costruire mediante questi elementi elementari un aggregato organizzato tale che, introdotto in un ambiente in cui essi abbondano, questo inizi a generare altri aggregati, ciascuno dei quali risulterà perfettamente identico all'originale. “La risposta è sì, e il principio in base al quale l'automa può essere realizzato è in stretta relazione col principio di Turing.” [von Neumann, 1951, p.163]

Von Neumann definisce allora un primo automa, *A*, che, fornito della descrizione *I* di un altro automa, è in grado di costruirlo assemblando le parti elementari. “Costruire” significa in questo caso selezionare e disporre i componenti in base a istruzioni che, come osserva, devono essere costituite da aggregati di parti elementari; quanto più l'oggetto da costruire è complesso, tanto maggiore sarà l'ampiezza della descrizione *I*. A questo si aggiunge un secondo automa, *B*, che può fare una copia di qualunque istruzione *I* ricevuta: un “*riproduttore*”, cioè una macchina che, dato un nastro con una descrizione, ne produce un'altra identica” Anche questo automa può produrre oggetti più grandi e più complicati di lui. Non v'è nulla di sorprendente in ciò. Poiché può solamente copiare, gli si dovrà fornire come entrata un oggetto esattamente delle dimensioni e della complessità dell'uscita” [von Neuamann, 1951, pp. 165-166 trad. it].

Il processo avviene in questo modo: *A* costruisce l'automa descritto dall'istruzione, *B* ne produce una copia, e *C* inserisce questa copia all'interno dell'automa appena costruito, separandolo infine come unità autonoma. La configurazione complessiva, che possiamo indicare come *D* (*A+B+C*), diventa autoriproduttiva quando è dotato dell'istruzione *I_D* che riceve descrive lo stesso *D*. In questo caso l'uscita del processo è un nuovo automa, identico a quello originario, capace a sua volta di ripetere la procedura., un automa capace di riprodursi da sé. Non vi è, secondo von Neumann, alcun paradosso: la copia di *I_D* avviene dopo che *D* è già stato formato e non lo altera, ma si aggiunge ad esso per costituire *E*. L'ordine dei passaggi è chiaro e consequenziale, e per questo “il processo è legittimo e corretto secondo le regole della logica” [Cfr. von Neumann, 1951, pp.164-165 trad. it.].

Su queste basi, von Neumann concluse la sua conferenza al simposio Hixon mettendo in relazione la teoria dell'automa autoriproduttivo con lo stato dell'arte delle ricerche sui geni. Alcune istruzioni possono essere interpretate come modelli genetici: un automa, ad esempio,

compie l'atto fondamentale della riproduzione, ossia la duplicazione del materiale genetico; variando le istruzioni, diventa possibile simulare fenomeni analoghi alla mutazione, nella maggior parte dei casi letali, ma talvolta compatibili con la prosecuzione della riproduzione, seppur con modificazioni nei caratteri. In altri casi, le istruzioni possono rappresentare geni che non solo si duplicano, ma stimolano anche la produzione di specifici "enzimi", cioè di ulteriori funzioni attive. Von Neumann riconosce che tali modelli non descrivono compiutamente un nuovo automa, ma forniscono piuttosto linee guida generali e indicazioni operative [Cfr.von Neumann, 1951, pp.165-166 trad. it.]. Tutto ciò lo condusse infine alla formulazione della soluzione al paradosso evolutivo da lui stesso enunciato: al di sotto di una certa soglia minima di complessità, un automa non può che generare copie più semplici e degenerative; al di sopra di essa, invece, diventa possibile la produzione di automi capaci di riprodurre progenie di pari o maggiore complessità, aprendo la via a un processo evolutivo non degenerativo. Come osserva Margaret Boden, la descrizione in termini informativi della replicazione biologica fu proposta da von Neumann "quando il meccanismo biologico dell'ereditarietà era ancora un mistero. Quando, cinque anni più tardi (nel 1953), fu scoperta la doppia elica e, qualche anno dopo, venne decifrato il codice genetico, si rivelò che la riproduzione biologica corrispondeva effettivamente alla descrizione astratta proposta da von Neumann". Il matematico ungherese definì la logica funzionale della replicazione. Quest'ultima concepita in termini computazionali, implicava che l'auto-organizzazione venisse considerata come una forma di calcolo, la quale richiede una soglia minima di complessità. Von Neumann intuì che una parte del sistema autoriproduttivo dovesse operare simultaneamente come istruzioni e come dati - vale a dire, come una autodescrizione [Boden, 2006, p.1268 trad. mia; Cfr. Boden, 1996, p.6]⁷⁷¹.

È da sottolineare come le parole finali al simposio Hixon di von Neumann indichino una sorta di programma di ricerca in questo senso:

Quanto si è detto non è che un primo passo grossolano verso la costruzione di una teoria sistematica degli automi. Esso rappresenta inoltre solo una particolare direzione d'indagine, cioè, come

⁷⁷¹ "The biological mechanism of heredity was still a mystery when von Neumann first described it in informational terms. When the double helix was discovered five years later (in 1953), and the genetic code decrypted some years after that, it turned out that biological reproduction did indeed fit von Neumann's abstract description. He thought of replication as a computational process, requiring a minimal level of computational complexity. Specifically, he realized that part of the self-replicating system must function both as instructions and as data-namely, as a self-description".

ho già detto, la direzione verso la formazione di un concetto rigoroso di complicazione. Il risultato di tale indagine è che la complicazione è probabilmente degenerativa ai livelli inferiori, cioè ogni automa che può produrre altri automi può produrne solo di meno complicati di sé. C'è tuttavia un livello minimo a cui questa caratteristica degenerativa cessa di essere generale. A questo punto diventano possibili automi che possono riprodursi e persino costruire oggetti più complessi di sé. Questo fatto, che la complicazione, come del resto l'organizzazione, al di sotto di un certo livello minimo sia degenerativa. e al di sopra di esso possa diventare autosostenentesi o addirittura crescente, svolgerà probabilmente un ruolo importante in ogni teoria futura in materia [von Neumann, 1951,p. 166 trad. it.].

Il modello cinematico fu poi approfondito nella quinta⁷⁷² lezione tenuta all'università dell'Illinois il 1949. I componenti elementari, da 10 a 15, qui vengono visti come analoghi dei neuroni, dei muscoli, del tessuto connettivo e di un produttore di energia metabolica. [ed. Burks, von Neumann, 1966, p.77]. L'automa cinematico anche qui doveva produrre altri automi e copie di sé stesso, ma a differenza della trattazione al simposio Hixon, "von Neumann voleva che la generazione della prole avvenisse a partire da un insieme di istruzioni, così da modellare in maniera più precisa il processo biologico di autoriproduzione" [Aspray, 1990, p.198 trad. mia]⁷⁷³. In questa lezione il matematico ungherese propone un modello di costruzione cinematica di una macchina di Turing,, capace di realizzare, a partire da componenti rigide elementari, sia un nastro binario che un "automa costruttore" capace di leggerne il nastro ed eseguirne le istruzioni [Cfr. ed. Burks, von Neumann, 1996, pp.83-87].

⁷⁷² Nella quarta lezione all'Università dell'Illinois dal titolo *The Role of High and Extremely High Complication*, von Neumann riprese l'analisi della complessità confrontando calcolatori elettronici e sistema nervoso. Stimò che il cervello umano fosse molto più complesso del computer: possiede un milione di volte più neuroni rispetto alle valvole di un calcolatore, e anche considerando la maggiore velocità di commutazione delle valvole, il vantaggio complessivo risultava a favore del sistema nervoso di circa 200 volte. Inoltre, il computer occupava un volume 10^9 volte maggiore, dissipava 10^9 volte più energia e risultava dal punto di vista termodinamico 200.000 volte meno efficiente nell'uso energetico per operazione binaria [Cfr. ed Burks, von Neumann, 1966 p.67]. Pur riconoscendo gli aspetti analogici presenti nei calcolatori digitali (soprattutto nelle interfacce di input/output), concluse che i metodi digitali sono indispensabili per affrontare situazioni complesse, mentre i meccanismi analogici possono operare solo semplificando e scomponendo i problemi in parti discrete [Cfr. ed Burks, von Neumann, 1966, pp.69-70]. Inoltre, viene in questa lezione ribadita la differenza tra automi biologici ed artificiali nella gestione dell'errore. In questa stessa lezione, von Neumann sottolineò come la differenza tra automi biologici e artificiali emerga in modo netto nella gestione dell'errore. Gli organismi naturali possiedono la capacità di sopravvivere anche in presenza di un'alta incidenza di errori, grazie a una flessibilità intrinseca e alla possibilità di sorvegliarsi e riorganizzarsi autonomamente. Ciò implica un'elevata autonomia delle parti, caratteristica evidente nel sistema nervoso umano, in cui le singole componenti possono riorganizzarsi e persino assumere il controllo in situazioni di emergenza. Questo livello di autonomia, tuttavia, può condurre anche a relazioni antagonistiche fra le parti, fenomeno del tutto assente negli automi artificiali e probabilmente connesso alla complessità del funzionamento degli organismi naturali. [Cfr. ed Burks, von Neumann, 1966, p.73].

⁷⁷³ "As originally envisioned in 1948, the kinematic automaton was designed to build copies from patterns, but by the time of the Illinois lectures, he wanted to build the offspring from a set of instructions in order to model more exactly the biological process of self-replication".

Il trattamento turinghiano, *digitale*, degli automi fornito da von Neumann in questa occasione è ammesso dallo stesso matematico ungherese, che dichiarava al suo uditorio:

Vedete, mi sto avvicinando molto all'espedito di Turing con gli automi universali, che prese avvio anch'esso da una descrizione formale generale di automi. Se si prendono quelle dozzine di elementi cui mi sono riferito in modo piuttosto vago e generale e se ne forniscono descrizioni esatte (cosa che potrebbe essere fatta in non più di due pagine stampate), si otterrà un linguaggio formale per descrivere gli automi in maniera univoca. Ora, ogni notazione può essere espressa in una notazione binaria, registrabile su un nastro perforato a canale singolo. Così, qualunque descrizione di automa potrebbe essere punzonata su un nastro. All'inizio, è meglio non utilizzare una descrizione dei pezzi e di come essi si incastrino, ma piuttosto una descrizione delle operazioni consecutive utilizzate per costruire l'automa [ed. Burks, von Neumann, 1966, p.83]⁷⁷⁴.

Tuttavia, malgrado l'interesse che il modello cinematico di automa poteva suscitare – che fu addirittura preso in considerazione dalla NASA negli anni '80 del Novecento per studiare la possibilità di fattorie lunari autoreplicanti [Cfr, Levy, 1992, pp.33-34] – questo lasciava insoddisfatto von Neumann. Questa teoria di autoriproduzione degli automi “presentava alcuni inconvenienti, poiché si trattava di macchine formate da un numero limitato di componenti situate nello spazio, per così dire, in un ambiente geometrico: il sistema descritto era capace di prendere queste componenti e di lavorare con esse”. “Era “troppo legato a considerazioni fisico-geometriche” [Israel, Millán Gasca, 2008, p.225]”.

Il modo di analizzare gli automi presentato al simposio Hixon e alle lezioni dell'Università dell'Illinois era “limitato a un tipo specifico di sistema fisico e lasciava inspiegati processi cruciali (per esempio, il riconoscimento, la presa e la saldatura delle parti)” [Boden, 2006, p.1269 trad. mia]⁷⁷⁵. L'ammissione è dello stesso von Neumann che nel corso della quinta lezione del 1949, intitolata *Re-evaluation of the problems of complicated automata-problems of hierarchy and evolution*, dichiara all'uditorio: “Assiomatizzando gli automi in questo modo, si è

⁷⁷⁴ “You see, I'm coming quite close to Turing's trick with universal automata, which also started with a general formal description of auto. If you take those dozen elements I referred to in a rather vague and general way and give exact descriptions of them (which could be done on two printed pages or less) you will have a formal language for describing automata unambiguously. Now any notation can be expressed as a binary notation, which can be recorded on a punched tape with a single channel. Hence any automaton description could be punched on a piece of tape. At first, it is better not to use a description of the pieces and how they fit together, but rather a description of the consecutive steps used in building the automaton”.

⁷⁷⁵ “Von Neumann himself, however, was dissatisfied with it. It was limited to a specific type of physical system, and it left crucial processes unexplained (for example, recognizing, picking up, and welding the parts)”

buttata via metà del problema, e potrebbe essere la metà più importante. Ci si è rassegnati a non spiegare come queste parti siano costituite da cose reali, in particolare, come siano fatte di particelle elementari effettive, o persino di molecole chimiche più complesse” [ed. Burks, von Neumann, 1966, p.77 trad. mia]⁷⁷⁶.

Stanislaw Ulam racconta nella sua autobiografia che su suo suggerimento, von Neumann abbandonò le considerazioni di natura fisica sugli automi per un approccio di natura logico-matematica:

Le nostre conversazioni abituali riguardavano la matematica o il suo nuovo interesse per una teoria degli automi. Queste conversazioni erano iniziate in modo sporadico e superficiale prima della guerra, in un periodo in cui simili argomenti quasi non esistevano. Dopo la guerra e prima della sua malattia avemmo molte discussioni su questi problemi. Gli proposi alcune delle mie idee sugli automi composti da celle disposte in una struttura simile a un cristallo [*crystal-like arrangements*] [Ulam, 1976, p.239 trad. mia]⁷⁷⁷.

Queste strutture, definite anche regolarità cristalline [*crystalline regularities*] o tassellazioni [*tilings*] [Aspray, 1990, p.203, Boden, 2006, p.1269], sono i *cellular automata*, gli *automi cellulari* [CA].

Questi ultimi dovevano costituire l’oggetto di un libro che avrebbe dovuto essere pubblicato dalla University of Illinois Press, ma i numerosi impegni di von Neumann ne rimandarono continuamente la stesura definitiva, cosicché il testo rimase incompiuto fino alla sua morte. Nel 1966 fu dato alle stampe dal suo collaboratore Arthur W. Burks con il titolo *Theory of Self-Reproducing Automata*, insieme a un altro manoscritto originale, anch’esso inedito e incompleto, intitolato *The Theory of Automata: Construction, Reproduction, Homogeneity* [Israel, Millán Gasca, 2008, p.225].

Un CA (*Cellular Automaton*, automa cellulare) è definito come un “array bidimensionale omogeneo di celle quadrate, ciascuna contenente lo stesso automa” [Aspray, 1990, p.203, trad. mia]⁷⁷⁸. Su suggerimento di Julian Bigelow e Herman Goldstine, von Neumann adottò una

⁷⁷⁶ “By axiomatizing automata in this manner, one has thrown half of the problem out the window, and it may be the more important half. One has resigned oneself not to explain how these parts are made up of real things, specifically, how these parts are made up of actual elementary particles, or even of higher chemical molecules”.

⁷⁷⁷ “Our usual conversations were either about mathematics or about his new interest in a theory of automata. These conversations had started in a sporadic and superficial way before the war at a time when such subjects hardly existed. After the war and before his illness we held many discussions on these problems. I proposed to him some of my own ideas about automata consisting of cells in a crystal-like arrangement”.

⁷⁷⁸ “They are homogeneous, two-dimensional array s of square cells”.

struttura bidimensionale anziché tridimensionale: le configurazioni a due dimensioni risultavano più semplici da gestire, e qualunque automa finito nello spazio cellulare tridimensionale può essere tradotto in una struttura equivalente all'interno dello spazio cellulare bidimensionale [Burks (ed.), von Neumann, 1966, p.105 n.24].

Qualsiasi cella può assumere, in un dato momento, lo stato ineccitabile (che rappresenta l'assenza di un neurone), uno dei venti stati quiescenti (a riposo, ma eccitabili), o uno degli otto stati sensibilizzati (eccitati). Le celle quiescenti possono rispondere a stimoli ordinari provenienti dalle celle adiacenti; le celle ineccitabili rispondono solo a stimoli speciali, che avviano la costruzione o la crescita di nuove celle. Lo stato di una data cella in un dato momento è determinato, secondo un insieme di regole di transizione, dagli stati di quella cella e dei suoi quattro vicini non diagonali in un certo momento precedente. All'istante 0 un automa cellulare ha un'assegnazione iniziale di celle costituita da una distribuzione finita di celle eccitate e quiescenti. L'autoriproduzione si verifica se l'assegnazione iniziale delle celle viene copiata in una diversa regione dell'insieme cellulare. [Cfr. ed. Burks, von Neumann, pp.132-156; Aspray, 1990, p.203].

“Era dunque possibile ottenere configurazioni che formavano un sistema analogo a un'entità viva, capace di muoversi e riprodursi, nel senso che poteva indurre altre cellule a entrare in uno stato analogo” [Israel, Millán Gasca, 2008, p.226]. In tal modo, il modello di automa cellulare⁷⁷⁹ diveniva una rappresentazione computazionale in uno spazio digitale della vita stessa. Dall'analisi delle componenti minimali del vivente inaugurata con la lettera a Wiener, si giunge ad un modello di vita artificiale.

⁷⁷⁹ Nel marzo 1953 von Neumann tenne le *Vanuxem Lectures* a Princeton. Nella quarta lezione presentò i suoi modelli di automi autoriproduttori: accanto al modello cinematico e a quello cellulare, introdusse il modello di soglia-eccitazione-affaticamento, concepito per simulare più da vicino il funzionamento del neurone. In questa formulazione, Ogni cella era dunque dotata di un meccanismo di attivazione, affaticamento e due periodi refrattari, riproducendo lo schema di eccitazione e recupero tipico dei neuroni umani. [Cfr. Burks, von Neumann, 1966, p. 96]. Seguivano il modello continuo, che tramite equazioni differenziali parziali intendeva rappresentare la diffusione ionica nel sistema nervoso, e il modello probabilistico, in cui le transizioni di stato erano determinate non in modo deterministico ma probabilistico, in continuità con le lezioni del Caltech sugli automi probabilistici, per esplorare mutazioni ed evoluzione [Cfr. Aspray, 1990, pp. 203–204]. La relazione fra questi modelli rifletteva quella tra computer analogici e digitali: il modello continuo si collocava sul versante analogico, mentre i modelli cellulare e probabilistico appartenevano al dominio discreto-digitale. Negli stessi anni Alan Turing, con *The Chemical Basis of Morphogenesis* (Turing, 1952), applicava anch'egli equazioni differenziali per spiegare l'emergere di strutture biologiche regolari (macchie, striature, segmenti) a partire da tessuti omogenei, introducendo il concetto di *morfogeni*. Ma mentre Turing utilizzava i modelli continui con finalità biologiche, per chiarire i meccanismi di auto-organizzazione della morfogenesi [Turing, 1952], von Neumann concepì il proprio modello continuo con finalità eminentemente computazionali, come parte di una teoria generale dell'autoriproduzione [Cfr. Boden, 2006, pp.1262-1264; Woolley et al., 2017, p.373-381]. John G. Kemeny (1926–1992), matematico e informatico, già assistente di Einstein a Princeton, contribuì a diffondere le idee di von Neumann sugli automi con *Man Viewed as a Machine* (Kemeny, 1955), pubblicato su *Scientific American*.

Si può osservare che il *design* degli automi cellulari di von Neumann rappresenta, almeno potenzialmente, un superamento dell'architettura seriale che porta il suo nome. ““il primo modello discreto e parallelo di calcolo nella storia formalmente dimostrato come un computer universale” [Berto, Tagliabue, 2023 trad. mia]⁷⁸⁰, “Ma”, come commenta Burks, ““nel progettare il suo automa autoriproduttore von Neumann non fece grande uso del potenziale parallelismo della sua struttura cellulare. Al contrario, il suo automa autoriproduttore funziona come un computer digitale seriale.” [ed. Burks, von Neumann, 1966, p.157 trad. mia]⁷⁸¹.

La macchina di Turing, che von Neumann aveva cercato di realizzare *in hardware* impiegando nella sua teorizzazione la metafora biologica della cibernetica – inglobate in sé la teoria dell'informazione e la digitalizzazione – diventata una modellizzazione delle forme strutturali della vita stessa⁷⁸², capace di calcolare qualsiasi funzione ricorsiva. Cioè una macchina universale [Berto, Tagliabue, 2023; Kari, 2003; Sarkar, 2000]⁷⁸³.

⁷⁸⁰ “Von Neumann’s CA was also the first discrete parallel computational model in history formally shown to be a universal computer”.

⁷⁸¹ “Thus von Neumann’s cellular structure allows for an indefinite amount of parallelism. But in designing his self-reproducing automaton von Neumann did not make much use of the potential parallelism of his cellular structure. Rather, his self-reproducing automaton works like a serial digital computer”.

⁷⁸² Il modello di automi cellulari autoriproduttori elaborato da von Neumann negli anni Quaranta e Cinquanta è considerato da Margaret Boden la radice storica dell’*Artificial Life* (*A-Life*), poiché rappresenta il primo vero tentativo di costruire in termini puramente computazionali un sistema capace di autoriproduzione e di evoluzione, anticipando così le linee di ricerca che verranno sviluppate decenni più tardi (Boden, 2006; 1996). Negli anni Cinquanta Nils Aall Barricelli (1912–1993) fu tra i primi a sperimentare l’evoluzione artificiale al calcolatore, conducendo al Princeton IAS computer simulazioni numeriche di popolazioni digitali autoriproduttori, capaci di mutazione e simbiosi, pubblicate poi nel 1962 (Barricelli, 1962; Cfr. Dyson, 2012; pp.261-280 trad. it.). Negli stessi anni John H. Holland (1929–2015), ispirato dalle intuizioni evolutive di von Neumann, sviluppò gli algoritmi genetici, formalizzati in *Adaptation in Natural and Artificial Systems* (1975), in cui mutazione, crossover e selezione permettevano a popolazioni di programmi di evolversi secondo principi darwiniani [Holland, 1962; 1975; Cfr. Boden, 2006, pp.1274-1278]. Negli anni Ottanta Christopher G. Langton (1948–2024), informatico statunitense, coniò il termine e definì l’*A-Life* come lo studio dei sistemi che mostrano i processi caratteristici della vita “così come potrebbe essere” [*life-as-it-could-be*], e non soltanto “così come la conosciamo” (Langton, 1988, p.1). A partire da questa impostazione, l’*A-Life* si è articolata in tre principali direzioni: la vita *in silico*, ossia simulata all’interno di computer ed ecosistemi artificiali; la vita *in hardware*, con robot e automi fisici; e la vita *in wetware*, basata su esperimenti di biologia sintetica e protocellule (Bedau, 2003). Come sottolineano Levy (Levy, 1992) e Boden [Boden, 1996], lo scopo è duplice: da un lato comprendere meglio i principi fondamentali che governano la vita naturale mediante modelli semplificati e controllabili; dall’altro esplorare i limiti teorici di ciò che può essere considerato vivente, aprendo la strada a nuove forme di organizzazione artificiale. In questo senso, l’intuizione di von Neumann rimane centrale: la sua astrazione dell’autoriproduzione in termini di automi cellulari continua a costituire il paradigma di riferimento per gran parte dell’*A-Life* contemporaneo. Una panoramica delle prospettive contemporanee dell’*A-Life* si trova in Dorin, Stepney (2024). Per una interessante analisi del legame dell’*A-Life* con i temi dei concetti fondamentali della cibernetica e dell’intelligenza artificiale si rimanda a Cordeschi, 1998, p.245-253.

⁷⁸³ Il modello originario di John von Neumann a 29 stati è stato poi perfezionato. Nei primi anni ’60 Edward F. Moore (1925–2003) e John Myhill (1923–1987) formularono il teorema del Garden of Eden, mostrando che un automa cellulare può presentare configurazioni che non hanno predecessori, rivelando connessioni profonde tra iniettività e suriettività nella dinamica dei CA [Myhill, 1963]. Nel 1966 Michael A. Arbib (1940–) pubblicò *Simple Self-Reproducing Universal Automata*, riformulando la complessa costruzione di von Neumann in forma più accessibile e più vicina a una concezione biologica (Arbib, 1966). La svolta concettuale e culturale avvenne nel 1970 con il *Game of Life* ideato da John Horton Conway (1937–2020), un CA bidimensionale con solo 2 stati

“Lo studio formalistico degli automi è un argomento che si colloca nell’area intermedia tra logica, teoria della comunicazione e fisiologia” [ed. Burks, von Neumann, 1966 p.91 trad. mia]⁷⁸⁴. In linea con l’analisi qui seguita, ciò riflette l’idea di una trattazione funzionalmente equivalente di ogni sistema dal punto di vista dell’informazione, indipendentemente dalla sua natura, alla luce della “riforma” del programma di ricerca cibernetico delineata da von Neumann nella missiva a Wiener del novembre 1946.

La possibilità di studio computazionale dell’informazione negli automi, biologici e artificiali – e la capacità eventuale di questo di affrontarne la complessità - emerge nella serie di domande metodologiche che von Neumann pose come guida della sua ricerca in questo campo:

- *Universalità logica* – Quando una classe di automi è in grado di eseguire tutte quelle operazioni logiche che possono essere svolte con mezzi finiti?
 - *Costruttibilità* – Qual è la classe di automi che può essere costruita a partire da “materie prime” da parte di un automa opportunamente definito?
 - *Universalità costruttiva* – Esiste un automa capace di costruire ogni altro automa?
 - *Autoriproduzione* – Esiste un automa in grado di costruire un automa del tutto identico a sé stesso?
 - *Evoluzione* – La costruzione di automi può evolvere verso tipi sempre più complessi?
- [ed. Burks, von Neumann, 1966 pp. 92; Aspray, 1990, p.204]⁷⁸⁵

Con il suo approccio logico-matematico von Neumann affrontò queste cinque questioni in modo sistematico. Alla prima, relativa all’universalità logica, rispose mostrando come il

(vivo/morto) e semplici regole di nascita e sopravvivenza che generavano dinamiche complesse ed emergenti. La sua fama fu amplificata dagli articoli di Martin Gardner (1914–2010) nella rubrica *Mathematical Games* di *Scientific American* (1970–71), che ne diffusero le configurazioni sorprendenti [Gardner, 1970]. La dimostrazione rigorosa della Turing-equivalenza del *Game of Life* fu formalizzata in seguito da Elwyn Berlekamp, John Conway e Richard Guy in *Winning Ways* (1982). Nel 1971, nella sua tesi dottorale, E. Roger Banks dimostrò che l’universalità computazionale poteva essere ottenuta anche con automi cellulari bidimensionali a numero di stati molto ridotto rispetto al modello originario di von Neumann [Banks, 1971]. Negli anni '80 e '90 Stephen Wolfram (1959–) intraprese un’analisi sistematica degli automi cellulari elementari unidimensionali (2 stati, raggio 1), classificandoli in quattro classi di comportamento (stabile, periodico, caotico, complesso) e introducendo il concetto di *edge of chaos* come zona intermedia in cui emergono fenomeni computazionalmente complessi (Wolfram 1986; 2002). In questo contesto la Regola 110, appartenente alla classe IV, si rivelò particolarmente importante – si tratta di un automa elementare unidimensionale con due soli stati (0/1) e raggio 1, le cui semplicissime regole locali generano tuttavia dinamiche ricche di strutture complesse – e nel 2004 Matthew Cook dimostrò che essa è Turing-universale [Cook, 2004], confermando che anche sistemi con regole minime possono esprimere tutta la potenza della computazione universale.

⁷⁸⁴ “The formalistic study of automata is a subject lying in the intermediate area between logic, communication theory, and physiology”.

⁷⁸⁵ La serie di domande segue la riformulazione fattane da William Aspray, in Aspray, 1990 p.204.

modello di macchina di Turing potesse essere riformulato in termini di automi cellulari, chiarendo così che questi ultimi erano in grado di eseguire tutte le operazioni logiche realizzabili con mezzi finiti [Cfr. ed. Burks, von Neumann, 1966, pp. 286-287; Aspray, 1990, p. 204].

Per quanto riguarda la costruttibilità e l'universalità costruttiva, von Neumann delineò un costruttore capace di generare altri automi a partire da stimoli che trasformavano celle quiescenti in eccitabili. Questo dispositivo fu successivamente esteso a un costruttore universale che, incorporato nella matrice bidimensionale del modello CA e dotato della descrizione di un automa su un "nastro" analogo a quello turinghiano, poteva costruire qualsiasi altro automa. Infine, egli mostrò come tale costruttore universale potesse essere progettato in modo da riprodurre se stesso [Cfr. ed. Burks, von Neumann, 1966, pp. 287 e 291].

Diverso fu il caso della quinta domanda, relativa all'evoluzione: per evitare le complicazioni del modello cinematico, von Neumann si limitò a osservare come i conflitti tra automi potessero dar luogo a dinamiche analoghe alla selezione naturale e come le modifiche delle descrizioni ereditate potessero fungere da mutazioni, senza tuttavia offrire una risposta definitiva [Cfr. Burks, ed., von Neumann, 1966, p.131].

L'amico e collaboratore di lunga data H. Goldstine ritiene che von Neumann "abbia forse considerato il suo lavoro sugli automi come il più importante [...] Esso non solo collegava il suo interesse iniziale per la logica con il suo successivo lavoro sulla neurofisiologia e sui computer, ma aveva anche la potenzialità di permettergli di dare contributi davvero profondi a tutti e tre i campi attraverso un unico apparato" [Goldstine, 1972, p.285 trad. mia]⁷⁸⁶.

I dettagli della teoria degli automi rimasero incompleti. von Neumann aveva progettato di tornare a quei manoscritti (curati e completati da Arthur Burks) al termine del suo incarico come commissario presso l'*Atomic Energy Commission*, Ma non poté farlo: fu colpito da un tumore al pancreas che lo portò alla morte nel 1957, prima che riuscisse a completare il progetto.

Tuttavia, lo studio degli automi costituì per lui la base per affrontare l'oggetto informazionalmente e fisiologicamente più complesso: il cervello umano. La complessità di quest'ultimo aveva infatti ispirato un programma di ricerca alternativo, fondato sull'analisi di

⁷⁸⁶ "Thinking about von Neumann's contributions, I am of the opinion that he perhaps viewed his work on automata as his most important one, at least the most important one in his later life. It not only linked up his early interest in logics with his later work on neurophysiology and on computers but it had the potentiality for allowing him to make really profound contributions to all three fields through one apparatus".

“microcomponenti” biologiche elementari, sviluppato in parallelo all’automa artificiale cui egli contribuì forse più di chiunque altro: il computer elettronico a programma memorizzato

Prima di passare a quest’ultimo sforzo intellettuale di von Neumann, c’è da sottolineare un punto che ne ribadisce, ancora una volta, la lungimiranza. L’integrazione della dimensione probabilistica della teoria dell’informazione nella logica degli automi artificiali - la gestione e la ricomprensione dell’*errore* nelle strutture computazionali - è un tema di scottante attualità negli sviluppi della *computer science*, dell’IA contemporanea e dell’apprendimento delle macchine. Nel loro ormai “classico” *Deep Learning* (Goodfellow et al., 2016), tre figure di riferimento mondiale nel campo dell’intelligenza artificiale contemporanea - Yoshua Bengio (premio Turing 2018 e uno dei tre *godfathers* del deep learning, insieme a Geoffrey Hinton e Yann LeCun), Ian Goodfellow (ideatore delle *Generative Adversarial Networks*, GAN) e Aaron Courville - osservano:

Molti rami dell’informatica si occupano principalmente di entità completamente deterministiche e certe. Un programmatore può di solito presumere in tutta sicurezza che una CPU eseguirà ogni istruzione macchina in modo impeccabile. Gli errori hardware si verificano, ma sono abbastanza rari da far sì che la maggior parte delle applicazioni software non debba essere progettata per tenerne conto. Considerato che molti informatici e ingegneri del software lavorano in un ambiente relativamente pulito e certo, può risultare sorprendente che il *machine learning* faccia ampio uso della teoria della probabilità. Ciò accade perché il *machine learning* deve sempre confrontarsi con quantità incerte e, talvolta, deve anche affrontare quantità stocastiche (non deterministiche). [Goodfellow et al., 2016, p. 54, trad. mia]⁷⁸⁷.

3. L’irragionevole complessità del cervello

L’ultimo sforzo di von Neumann intorno alla teoria degli automi fu il lavoro per le *Silliman Lectures* dell’Università di Yale, a cui fu invitato a inizio 1955 per la primavera del 1956. Nominato da pochi mesi commissario dell’*Atomic Energy Commission* (AEC) dal presidente Eisenhower e già colpito dai sintomi del cancro che lo avrebbe portato alla morte, decise di

⁷⁸⁷ “Many branches of computer science deal mostly with entities that are entirely deterministic and certain. A programmer can usually safely assume that a CPU will execute each machine instruction flawlessly. Errors in hardware do occur, but are rare enough that most software applications do not need to be designed to account for them. Given that many computer scientists and software engineers work in a relatively clean and certain environment, it can be surprising that machine learning makes heavy use of probability theory. This is because machine learning must always deal with uncertain quantities, and sometimes may also need to deal with stochastic (non-deterministic) quantities”.

concentrare tutte le sue energie in questo progetto, rinunciando a ogni altro impegno scientifico. Sua moglie Klara ha lasciato un resoconto vivido ed emozionante dei suoi sforzi eroici: lavorava instancabilmente al manoscritto anche quando ormai era costretto sulla sedia a rotelle, chiedendo di ridurre le conferenze a pochi giorni pur di riuscire a portarle a termine. All'inizio del 1956 fu però chiaro che non avrebbe potuto sostenerle, e, ricoverato di nuovo, si portò in ospedale le carte delle lezioni nella speranza che potessero essere completate da altri. Non riuscì mai né a terminarle né a tenerle. Di quella serie rimangono due testi parziali, confluiti nel volume postumo *The Computer and the Brain* [von Neumann, 1958], che raccoglie la sua ultima riflessione comparativa sul funzionamento dei calcolatori e del cervello umano [Klara von Neumann, 1957; Aspray, 1990, p.206; Bhattacharya, 2021].

Se la modellizzazione neuronale di McCulloch-Pitts non fu accolta favorevolmente dalla comunità neurofisiologica, quest'ultimo lavoro di von Neumann "viene ancora letto oggi dagli studiosi delle scienze biomediche e il suo influsso ha cambiato radicalmente l'atteggiamento verso l'approccio che in esso viene propugnato" [Israel, Millán Gasca, 2008, p.227]. L'ultimo, incompiuto, sforzo intellettuale di von Neumann era presentare "uno dei confronti più sofisticati mai tracciati tra computer e cervelli, con la speranza che una nuova disciplina, la teoria degli automi, si sviluppasse presto per spiegarli entrambi" [Piccinini, 2003⁷⁸⁸].

L'introduzione di von Neumann chiarisce metodo con cui intende affrontare il problema: "Si tratta di un approccio alla comprensione del sistema nervoso dal punto di vista di un matematico" [von Neumann, 1958, p.53 trad.it. 2021]⁷⁸⁹.

Da questa analisi condotta dalla prospettiva matematica, lo scienziato ungherese mette in evidenza quanto emerso dallo studio della teoria degli automi, e cioè che "rimarranno sempre in primo piano nel discorso gli aspetti logici e statistici". Questi ultimi due dovrebbero costituire "gli strumenti di base di una «teoria dell'informazione» il cui oggetto di studio sarà l'intero corpo di esperienze cresciuto intorno alla progettazione, valutazione e programmazione degli automi logico-matematici complessi". Ossia, le macchine calcolatrici che von Neumann aveva contribuito a concepire e realizzare [von Neumann, 1958, p.53 trad.it. 2021].

⁷⁸⁸ "Von Neumann was one of the top experts in all aspects of computing-hardware, programming, numerical analysis, and computability theory-and one of the most rigorous minds ever to discuss the computational organization of brains. His last book presents one of the most sophisticated comparisons ever made between computers and brains, with the hope that a new discipline, automata theory, would soon develop to explain them both".

⁷⁸⁹ In questo lavoro impiego due traduzioni italiane di *The Computer and the Brain*. La prima è quella contenuta in Somenzi, Cordeschi, 1994; la seconda è quella di Bartesaghi, 2021. Distinguerò indicando la data della traduzione nel a quale delle due di volta in volta farò riferimento.

La teoria degli automi, come riconosciuto dallo stesso matematico di Budapest è ancora largamente incompiuta, “purtroppo, ciò che abbiamo a disposizione in questo momento – e a cui devo far riferimento – può ancora essere descritto soltanto come un «corpo di esperienze» sviluppato in modo imperfetto e scarsamente formalizzato”. Ma sulla base di questa metodologia embrionale, von Neumann sottolinea il suo “obiettivo primario”, sebbene implicito. “Ho il sospetto” dichiara von Neumann “che uno studio matematico più approfondito del sistema nervoso – «matematico» nel senso prima descritto – influenzerà la nostra comprensione di alcuni aspetti della stessa matematica coinvolta. In effetti, potrà modificare il modo in cui guardiamo la matematica e la logica stesse” [von Neumann, 1958, p.54 trad.it. 2021].

La prima parte del testo è dedicata all’analisi dei calcolatori, e “offriva una chiara panoramica divulgativa dei principi del calcolo moderno” [Aspray, 1990, p.208 trad. mia]⁷⁹⁰. Dopo aver analizzato le differenze tra il principio di calcolo analogico e quello numerico/digitale [von Neumann, 1958, pp.108-110 trad.it 1994], von Neumann mette in evidenza il carattere “misto” che possono assumere le macchine calcolatrici, di solito considerate puramente digitali [von Neumann, 1958, pp.118-120 trad.it 1994].

L’analisi passa poi alle varie forme di controllo: “Una macchina di calcolo, oltre ad avere la capacità di eseguire singolarmente le operazioni di base, deve essere in grado di eseguirle in una ben determinata successione o, meglio, in accordo a uno schema logico entro il quale esse possano generare la soluzione al problema matematico che rappresenta lo scopo effettivo del calcolo in esame” [von Neumann, 1958, p.65 trad. it. 2021].

Nella rassegna delle varie forme di controllo logico – in cui von Neumann ribadisce la “metafora cibernetica” di identificazione di componenti artificiali e organi biologici - ritroviamo i principi che abbiamo riscontrato incorporati nelle varie macchine che abbiamo preso in esame, dal *Differential Analyzer*, a quelle *special purpose* come Colossus o l’ENIAC che la realizzavano mediante configurazione di fili e prese mediante interfacce esterne o nastro fino al programma memorizzato per le macchine *general purpose*.

Nelle tradizionali macchine analogiche, il sequenziamento delle operazioni era assicurato dalla presenza di un numero sufficiente di organi – integratori e differenziali – collegati tra loro in modo da costituire una replica fedele del calcolo da eseguire. Nei primi modelli questi collegamenti erano meccanici, realizzati mediante ingranaggi, mentre nelle

⁷⁹⁰ “The first lecture, on the computer, gave an articulate popular overview of the principles of modern computing”.

macchine successive questi furono sostituiti da dispositivi elettrici e da interfacce esterne. In ogni caso, però, la configurazione restava immutabile per l'intera durata del problema: "Tuttavia, in tutti i tipi di macchina il set-up rimaneva fisso per l'intera durata del problema" [von Neumann, 1958, p.65 trad. it. 2021]

Un deciso passo in avanti fu rappresentato dal "controllo mediante interfaccia esterna": un pannello frontale con ingressi e connessioni controllate elettromeccanicamente, azionati a loro volta da impulsi elettrici provenienti, ad esempio, da nastri perforati, nelle forme di controllo "mediante nastro". [von Neumann, 1958, p.66 trad. it. 2021]. Qui abbiamo precedentemente messo in evidenza la relazione *simbiotica* tra le configurazioni potenziali della macchina e l'operatore umano per svolgere il compito computazionale richiesto.

Nelle macchine digitali si affermò fin da subito il principio secondo cui vi "è presente di norma un solo organo per ciascuna operazione di base" [von Neumann, 1958, p.67 trad. it. 2021]. A differenza delle macchine analogiche – che spesso duplicavano gli organi a seconda delle esigenze del problema – questo assetto risultava più ordinato ed economico, anche se si trattò "un fatto storico piuttosto che di una necessità intrinseca" - in linea di principio anche le macchine analogiche avrebbero potuto funzionare con un singolo organo e con sistemi di controllo logico simili a quelli digitali [Cfr. von Neumann, 1958, p.67 trad. it. 2021].

Questo principio, però, porta con sé una conseguenza cruciale: "richiede, tuttavia, che venga garantita la presenza di un numero maggiore di organi che possano essere utilizzati per memorizzare passivamente i risultati dei vari calcoli intermedi e parziali" [Cfr. von Neumann, 1958, p.68 trad. it. 2021]. Se un organo può svolgere una sola operazione alla volta, i risultati intermedi devono essere conservati altrove. Diventa quindi indispensabile introdurre organi di memoria in grado di registrare un numero, cancellarne uno precedente e ripeterlo quando richiesto. "Un tale oggetto è chiamato «registro di memoria»; la totalità di questi organi è chiamata «memoria» e il numero di registri in una macchina è detta «capacità di memoria»" [Cfr. von Neumann, 1958, p.68 trad. it. 2021].

Il nesso fra controllo logico e memoria è decisivo: solo coordinando le operazioni con l'accesso a una memoria centrale diventa possibile realizzare quello che sarà il cuore dell'architettura moderna del computer, lo *stored program principle*, in cui istruzioni e dati vengono trattati come informazioni dello stesso tipo e conservati nello stesso organo.

Dopo aver trattato il "controllo per punti sequenza", cioè mediante organi collegati alle "operazioni di base del sistema", che le azionano e sono connessi "ai registri di memoria deputati a fornire gli ingressi numerici per tale operazione e all'organo che dovrà ricevere i

risultati in uscita” [Cfr. von Neumann, 1958, pp. 68-71; trad. it. 2021], von Neumann passa alla trattazione del “controllo memorizzato”. Qui i punti sequenza sono sostituiti da “comandi”. La macchina numerica/digitale processa, come realizzazione della macchina di Turing, in input e output *numeri* in un’equivalenza sostanziale tra dati e programmi. “Nella gran parte delle forme in cui viene realizzato questo schema un comando è fisicamente equivalente a un numero [...] Quindi in una macchina decimale è semplicemente una sequenza di cifre decimali” [von Neumann, 1958, p.71 trad. it. 2021]. I computer per von Neumann sono “macina-numeri” [*numbers crunchers*] [Piccinini, 2003]. Questa interpretazione dei calcolatori può essere a tutta prima “sconcertante”:

Per molti scopi pratici, il modo migliore di interpretare i marcatori manipolati dai computer può ben essere quello di considerarli come rappresentazioni di numeri. Tuttavia, sono gli utenti a leggerli in questo modo: nessuna proprietà fisica della macchina rende necessaria tale interpretazione. I marcatori possono essere interpretati come rappresentazioni di cose diverse dai numeri, o anche di nulla; che gli utenti li leggano come numeri o come qualcos’altro non fa alcuna differenza per il funzionamento della macchina [Piccinini, 2003 trad. mia]⁷⁹¹.

La GOFAI ha interpretato i computer (e la mente), invece, come *symbol crunchers*, manipolatori di simboli secondo regole formali, ma questi non devono essere necessariamente cifre numeriche. “Certamente von Neumann, eminente logico qual era, conosceva perfettamente la distinzione tra sistemi simbolici e le loro interpretazioni, ma per qualche ragione non la discusse in questo contesto. Forse perché era consapevole che anche i numeri potevano essere visti come rappresentazioni di altre entità, come quantità fisiche o istruzioni per il computer” [Piccinini, 2003 trad. mia]⁷⁹². A mio avviso, la considerazione esclusivamente numerica che von Neumann dà del computer – e, contestualmente, del cervello – è da legare

⁷⁹¹ “Von Neumann’s explanation of computers, especially digital computers, contains a puzzling aspect. For many practical purposes, the best way to interpret the markers manipulated by computers may well be that they represent numbers. However, it is users who read markers in this way: no physical property of the machine necessitates this interpretation. The markers can be interpreted as representing things other than numbers, or nothing at all; whether users read markers as representing numbers or anything else makes no difference to the functioning of the machine”.

⁷⁹² “Indeed, digital computers are often described not as number crunchers but symbol crunchers, meaning that the markers are symbols that can represent a variety of things. According to this variant picture, computers do manipulate symbols according to formal rules, but the rules need not be interpreted as arithmetical operations, and the symbols need not be seen as numerals. Surely von Neumann, eminent logician that he was, knew the distinction between symbolic systems and their interpretations perfectly well, but for some reason he did not discuss it in this context. Perhaps this is because he was aware that numbers themselves could be seen as representing other entities, such as physical quantities or instructions for the computer”.

all'importanza data dal matematico ungherese al lavoro sui problemi di computabilità di Turing e delle sue macchine universali – e dalla loro derivazione gödeliana – che von Neumann voleva tradurre *in hardware* nei suoi calcolatori⁷⁹³.

Tornando all'analisi di von Neumann della natura dei “comandi”, questi specificano l'operazione di base da eseguire, gli indirizzi dei registri di memoria da cui prelevare gli input e quelli in cui collocare gli output. Inoltre “ogni comando deve determinare il suo successore” [von Neumann, 1958, p.72 trad. it. 2021]. Poiché “un comando è di solito fisicamente equivalente a un numero [...] il modo più naturale per memorizzarlo – nel corso del problema al cui controllo esso partecipa – è quello di collocarlo in un registro di memoria” [von Neumann, 1958, p.73 trad. it. 2021].

Qual è la differenza tra controllo a interfaccia elettromeccanica e quello così delineato da von Neumann? Nel secondo caso i comandi si *dematerializzano*. Nel primo i punti di sequenza “erano oggetti fisici reali e i loro collegamenti elettrici traducevano la natura del problema. Qui i comandi sono enti ideali immagazzinati nella memoria ed è pertanto il contenuto di questo particolare segmento di memoria a esprimere la natura del problema. Di conseguenza, questa modalità di controllo viene chiamata «controllo memorizzato»” [von Neumann, 1958, p.73 trad. it. 2021].

Questa dematerializzazione muta profondamente il “modus operandi” della macchina (e dell'utente) in base al controllo memorizzato. Poiché i comandi che regolano il funzionamento sono immagazzinati nella memoria, “si raggiunge un livello di flessibilità superiore rispetto a qualsiasi altra precedente modalità di controllo”. La memorizzazione del *programma* – del set di istruzioni – in forma numerica, fa sì che la macchina “in questo modo diventa capace di modificare il contenuto della memoria”. In questo modo aumenta la capacità della macchina di elaborare soluzioni complesse. Cambia anche il modo di interazione con il calcolatore, dall'inserimento manuale di spinotti e cavi alla programmazione. “Il problema da risolvere, l'intenzione dell'utente – viene comunicato alla macchina «caricandolo» in memoria” [von Neumann, 1958, p.74].

La trattazione dell'automa “computer” viene poi completata da von Neumann – dopo un'analisi dei dettagli tecnici e ingegneristici - con la trattazione degli organi che compongono

⁷⁹³ La scelta di von Neumann per una interpretazione esclusivamente in termini di manipolazione di cifre può anche essere letta come conseguenza del suo impiego del calcolatore nei numerosi problemi di analisi numerica – dalla meteorologia all'idrodinamica a quelli che verranno poi studiati attraverso il metodo Monte Carlo – in cui il matematico nato a Budapest era interessato. Sugli interessi di von Neumann intorno al tema si veda Aspray, 1990, p.95-172 e Goldstine, 1972, pp.286-299.

la “macchina digitale moderna”: Queste “sono costituite da organi «attivi» e da organi che hanno la funzione di «memoria».” (tra questi vengono inclusi quelli di input e di output) [von Neumann, 1958, p.84 trad. it. 2021]. Gli organi attivi comprendono, da un lato, quelli che eseguono le operazioni logiche fondamentali – rilevando coincidenze e anticoincidenze o combinando impulsi – e, dall’altro, quelli che rigenerano i segnali, ripristinandone l’energia, amplificandoli o riportandoli a una forma e durata standardizzata entro determinate tolleranze. [Cfr. von Neumann, 1958, p.84 trad. it. 2021].

Qui viene ribadita l’analogia che strutturava il framework teorico dell’EDVAC. Le componenti del calcolatore sono trattate in modo biologistico, al pari del loro equivalente funzionale, le parti del sistema nervoso. L’impiego della metafora cibernetica da parte di von Neumann “gli permise da un lato di paragonare l’unità aritmetica del calcolatore al neurone e, dall’altro, di interrogarsi su dove sia localizzata la memoria nel sistema nervoso umano” [Aspray, 1990, p.208 trad. mia]⁷⁹⁴

Nella seconda parte del lavoro von Neumann, può, sulla base dell’analisi svolta, affrontare il suo obiettivo. la “descrizione dell’altro termine di paragone, il sistema nervoso umano”, analizzando “le analogie e le differenze tra questi due tipi di «automi».”. Se le analogie, riconosce il matematico ungherese sono note – e cioè il carattere digitale e quelle derivanti dalla trattazione “cibernetica”, informazionale e computazionale delle componenti biologiche con quelle artificiali – “vi sono però anche differenze, e non solo per quanto attiene aspetti piuttosto evidenti quali le dimensioni e le velocità ma anche aspetti che si trovano più in profondità e che coinvolgono i principi di funzionamento, di controllo e di organizzazione complessiva dei due sistemi” [von Neumann, 1958, p.95 trad. it. 2021].

L’esame di von Neumann dell’automa biologico inizia con una descrizione della struttura neurofisiologica del neurone - la “componente di base” del sistema neurofisiologico - della natura dell’impulso nervoso e dei processi di stimolazione [von Neumann, 1958, pp.96-99 trad. it. 2021]. Questi ultimi sono trattati come fenomeni di natura “digitale”: Gli impulsi nervosi possono chiaramente essere considerati come dei segnali a due valori [...]: l’assenza dell’impulso rappresenta un valore (per esempio il numero binario 0), mentre la presenza dell’impulso rappresenta l’altro valore (per esempio il numero binario 1). [...] Essa deve allora

⁷⁹⁴ “Rather than give the standard account of the logical organization of the computer, von Neumann presented it in terms of active and memory elements, which enabled him to compare the arithmetic unit of the computer with the neuron and at another to ask where memory is located in the human nervous system”.

essere interpretata come un segnale (un numero binario 0 oppure 1) avente una funzione logica specifica” [von Neumann, 1958, p.129 trad. it 1994].

Tuttavia, già nella sua prima riflessione sulla natura del cervello, von Neumann osserva che il suo funzionamento è “*prima facie* digitale” [von Neumann, 1958, p. 96; trad. it. 2021], e tale caratteristica sembra confermata dalle considerazioni appena svolte sulla struttura degli impulsi nervosi. Egli aggiunge però, in continuità con i diversi passaggi in cui, nella trattazione degli automi, aveva sottolineato la natura mista – digitale e analogica – delle componenti biologiche e artificiali, che “la descrizione precedente contiene alcune idealizzazioni e semplificazioni [...]. Una volta che queste ultime siano prese in considerazione, il carattere digitale del sistema nervoso non emerge più in modo altrettanto chiaro e inequivoco” [von Neumann, 1958, p. 100; trad. it. 2021].

Queste differenze si approfondiscono nel confronto tra gli organi del cervello e quelli del calcolatore in termini di *dimensioni*; *dissipazione di energia* e *velocità*.

Dal punto di vista delle dimensioni, il paragone “vede i componenti naturali superare di un fattore di circa $10^8 \div 10^9$ i corrispondenti componenti artificiali”, e per quanto riguarda il volume – per calcolarne il numero – “i componenti naturali sono superiori a quelli artificiali per fattori come 10^8 o 10^9 ” [von Neumann, 1958, p. 106 trad. it. 2021; p.132 trad. it. 1994].

Riguardo all’efficienza energetica, il neurone è enormemente superiore: dissipa circa 10^{-7} watt contro i 5–10 watt di un tubo a vuoto e i 10^{-5} watt di un transistor. Ciò significa che è da 10^5 a 10^6 volte più efficiente dei componenti artificiali. [von Neumann, 1958, pp.132-133 trad. it. 1994].

Per quanto attiene il confronto intorno alla velocità, “esiste un fattore di circa $10^4 \div 10^5$ a favore dei componenti artificiali rispetto a quelli naturali” [von Neumann, 1958, p.106 trad. it. 2021]. Quest’ultima caratteristica ci fornisce un indizio importante sulle differenze di design tra l’automa biologico e quello artificiale:

von Neumann stima che i neuroni siano circa 10^5 volte più lenti rispetto a valvole o transistor per quanto riguarda il tempo richiesto a completare un’operazione logica di base [...] La conclusione è inevitabile. Se il cervello fosse un computer digitale con architettura di von Neumann, al confronto con una macchina sarebbe condannato a essere una tartaruga computazionale [Churchland, Churchland, 2000, p.37 trad. it. 2021].

Ma, è lecito chiedersi a questo punto, l'approfondimento della metafora cibernetica a che conclusioni architetture porta riguardo all'organizzazione degli automi?

Il matematico ungherese fornisce una prima risposta in relazione alle sue osservazioni sul rapporto numero/velocità delle componenti:

Quindi è prevedibile che un automa naturale di grandi dimensioni ed efficientemente organizzato (come il sistema nervoso umano) tenderà a raccogliere il maggior numero possibile di elementi logici (o informativi) simultaneamente e a elaborarli simultaneamente, mentre un automa artificiale di grandi dimensioni ed efficientemente organizzato (come una grande macchina di calcolo moderna) è più probabile che esegua operazioni in successione – una alla volta o comunque non troppe alla volta. Cioè, *i grandi ed efficienti automi naturali è molto probabile che siano altamente paralleli*, mentre i grandi ed efficienti automi artificiali tenderanno a esserlo meno e a essere invece molto più seriali [von Neumann, 1958, p.107 trad. it. 2021 enfasi mia].

L'analisi di von Neumann conduce a un ulteriore aspetto che ha attraversato tutto il corso della nostra analisi. L'analogia funzionale dei componenti biologici con quelli artificiali, visti come digitali, si è basata sul criterio, ritenuto sufficiente per realizzare le operazioni booleane, della *soglia*: infatti, se un neurone richiede la stimolazione simultanea di due impulsi per generare una risposta, esso si comporta come un organo logico "e" (AND), realizzando l'operazione di *congiunzione*, poiché risponde solo quando entrambi i suoi ingressi sono attivi. Se invece è sufficiente l'arrivo di un solo impulso per attivarlo, il neurone funziona come un organo logico "o" (OR), realizzando l'operazione di *disgiunzione*, dal momento che risponde quando almeno uno dei due ingressi è attivo [Cfr. von Neumann, 1958, pp.124-135 trad. it. 1994]. Ma questa "però è una semplificazione teorica. I neuroni, in realtà, non sono in genere organizzati così semplicemente per ciò che concerne la loro posizione nel sistema" [Cfr. von Neumann, 1958, p.135 trad. it. 1994].

L'organizzazione neuronale per implementare le funzioni logiche deve allora tenere conto di un altro tipo di organizzazione delle componenti e di elaborazione delle informazioni: le *sinapsi*. Se come notato poco fa l'automa biologico, tenderà a raccogliere "il maggior numero possibile di elementi logici (o informativi) simultaneamente e a elaborarli simultaneamente", conclude von Neumann, "può essere benissimo che alcune combinazioni di impulsi stimolino un dato neurone non semplicemente in virtù del loro numero, ma anche in virtù dei rapporti spaziali delle sinapsi cui pervengono" [von Neumann, 1958, p.135 trad. it 1994].

La valutazione del tipo e del numero di componenti deve considerare, secondo von Neumann, che “la cellula nervosa è più che un singolo organo attivo elementare e che ogni tentativo di calcolare il numero di tali organi deve tener presente questa considerazione”. Nel caso in cui la l’attivazione della cellula nervosa dipende da una combinazione sinaptica, “allora un conto degli organi attivi elementari che abbia qualche significato deve essere un conto di sinapsi piuttosto che di cellule nervose” [von Neumann, 1958, p.138 trad. it. 1994].

Le ultime considerazioni mettono in questioni un punto chiave dell’architettura di von Neumann e della considerazione logica dei suoi organi (architrave del principio del programma memorizzato): il problema della *memoria*.

Nel computer concepito secondo l’architettura di von Neumann, che in questo design seriale trova la sua realizzazione (ad esempio nell’EDVAC), la memoria costituisce il nucleo centrale come deposito comune di dati e istruzioni. Essa è organizzata in forma gerarchica e separata dal controllo logico, rendendo così possibile l’elaborazione sequenziale. Nel sistema nervoso, invece, la memoria, intesa come “componente” o “sottosistema del sistema nervoso centrale”, e la sua stessa localizzazione restano, secondo von Neumann, del tutto ignote: al di là di supposizioni induttive fondate sulla sua importanza nel calcolatore, ne “sappiamo quanto i greci, i quali supponevano che la mente fosse situata nel diaframma”. Non vi sono prove certe di dove essa sia situata, se sia un “organo separato”, “un insieme di determinate parti di organi già noti” o se “potrebbe benissimo avere a che fare col meccanismo genetico del corpo”. L’unica base, a cui la costruzione dei computer ha contribuito, “è che deve essere una memoria di grande capacità e che è difficile immaginare come potrebbe funzionare un automa così complicato come il sistema nervoso umano se non disponesse di una memoria” [von Neumann, 1958, p. 139; trad. it. 1994].

Dopo aver messo in evidenza queste difficoltà, von Neumann prova ad affrontare la questione della quantificazione della capacità di memoria degli automi, e lo fa servendosi della “teoria dell’informazione di Shannon”. “Una memoria può ritenere una certa quantità massima di informazione e l’informazione può essere trasformata in un aggregato di numeri binari, i cosiddetti *bit*” [von Neumann, 1958, p.139 trad. it. 1994].

Per spiegare il metodo, von Neumann, impiega come paragone esplicativo il calcolatore. Una memoria che contiene 1000 numeri decimali di otto cifre ha una capacità di $1000 \times 8 \times 3,32 \approx 2,66 \times 10^4$ bit, poiché ogni cifra decimale equivale a circa $\log_2 10 \approx 3,32$ bit. Analogamente, l’alfabeto dattiloscritto con 88 simboli possibili corrisponde a $\log_2 88 \approx 6,45$ bit per carattere: una memoria che contiene 1000 lettere vale quindi circa 6450 bit. Allo stesso modo, anche

figure, colori o altre informazioni possono essere espresse in bit a seconda della precisione della rappresentazione. Applicando lo stesso criterio al sistema nervoso, tenta allora una stima approssimativa: ipotizza che un recettore standard riceva circa 14 impulsi digitali al secondo, equivalenti ad altrettanti bit, e che ciascuna delle circa 10^{10} cellule nervose possa, in condizioni appropriate, agire come recettore. Ne deriva un flusso informativo complessivo di 14×10^{10} bit al secondo. Proiettando questo valore sull'intera durata di una vita media di 60 anni, pari a circa 2×10^9 secondi, il risultato è una capacità complessiva di circa $2,8 \times 10^{20}$ bit. [Cfr. von Neumann, 1958, pp.139-140 trad. it. 1994].

“Tuttavia”, conclude von Neumann, “non sembra che questa cifra sia spropositata tenendo conto di quanto già osservato prima per ciò che riguarda il numero di organi attivi elementari cervello rispetto a quello dei calcolatori” [von Neumann, 1958, p.140 trad. it. 1994]

Riguardo alla possibile localizzazione, von Neumann passa in rassegna varie ipotesi sulla localizzazione fisica della memoria nel sistema nervoso. Una possibilità suggerita dalla letteratura scientifica neurofisiologica è che essa possa risiedere nella variabilità delle soglie di stimolazione delle cellule nervose, che cambierebbero in funzione della loro attività precedente. Un'altra è che riguardi le connessioni stesse tra le cellule: l'uso abituale di un assone potrebbe renderlo meno efficiente o, al contrario, abbassarne la soglia e facilitare la trasmissione. Viene poi richiamata la memoria genetica, cioè i cromosomi e i geni come elementi che conservano e trasmettono informazioni, e infine altre forme più speculative, legate alla composizione chimica di alcune zone del corpo [Cfr. von Neumann, 1958, pp.140-141 trad. it. 1994].

In paragone con i calcolatori come l'ENIAC, von Neumann sottolinea come i sistemi nervosi di cellule che si stimolano a vicenda possono essere visti come memorie composte di elementi attivi, analoghe ai “flip-flop” delle prime macchine a valvole. Queste soluzioni però risultano estremamente costose e poco plausibili nel caso del sistema nervoso, che verosimilmente si avvale di forme di memoria fondate su principi diversi, come accade nei calcolatori moderni (memorie elettrostatiche, magnetiche, a ritardo acustico, ecc.). [Cfr. von Neumann, 1958, pp.141-142 trad. it.1994].

Tuttavia, malgrado l'importanza postulata della memoria, degli organi che la compongono e della loro natura e localizzazione, non si riesce a gettare luce su questi temi.

Sembra che questi aspetti della questione siano molto importanti per la comprensione della struttura del sistema nervoso, e che essi siano ancora fondamentalmente insoluti. Conosciamo gli organi

attivi elementari del sistema nervoso (le cellule nervose), e ci sono tutte le ragioni per credere che una gran parte della memoria sia connessa a questo sistema. Ma noi non sappiamo nel modo più assoluto che tipo di entità fisiche siano i componenti fondamentali di questa memoria [von Neumann, 1958, p. 142 trad. it. 1994].

Un'ulteriore difficoltà nello studio della "sottosezione mnestica" del sistema nervoso riguarda il suo carattere misto, più volte palesatosi nella nostra analisi della teoria degli automi, che alterna continuamente aspetti digitali e analogici. Gli impulsi nervosi hanno natura digitale, ma possono controllare processi analogici come contrazioni muscolari o secrezioni chimiche, i quali a loro volta generano nuovi impulsi digitali. Questo passaggio dal digitale all'analogico e viceversa può ripetersi più volte nello stesso processo. [Cfr. von Neumann, 1958, pp.142-143 trad. it.1994].

Lo stesso schema si riscontra anche nei meccanismi genetici: "I geni", afferma von Neumann, "di per sé sono parte di un sistema digitale di componenti. I loro effetti tuttavia consistono nello stimolare la formazione di determinate sostanze chimiche, e precisamente di enzimi che sono caratteristici del singolo gene; si tratta quindi di effetti di tipo analogico" [von Neumann, 1958, p. 143 trad. it. 1994].

Dalla trattazione della memoria emerge un ulteriore elemento di distinzione. Non solo l'architettura dell'automa neurofisiologico si delinea come *parallela* e strutturata secondo un principio di implementazione booleana fondato su componenti elementari differenti dal neurone, le *sinapsi*, ma anche la *memoria* si configura con capacità, natura e localizzazione diverse rispetto a quelle previste nel design dell'automa artificiale. Resta tuttavia aperta una questione fondamentale: la *struttura logica* del sistema nervoso.

Questi problemi riguardano "i rispettivi ruoli della logica e dell'aritmetica nel funzionamento di un qualsiasi automa complesso, e, nel caso specifico, del sistema nervoso" [von Neumann, 1958 p.131 trad.it. 2021]. "Qualsiasi automa artificiale che è stato costruito per uso umano", nota von Neumann, "possiede di norma una parte logica e una parte aritmetica". Il progettista, nel costruire un automa per affrontare un problema complesso, ad esempio di natura ingegneristica o fisiologica, ricorre tanto a uguaglianze e disuguaglianze numeriche quanto a condizioni puramente qualitative che combinano risposte e non-risposte. "Alla luce delle nostre abitudini di pensiero e delle sue forme di espressione, ciò si può spiegare con il fatto che ci risulta assai difficile esprimere qualsiasi situazione davvero complicata senza ricorrere a formule e numeri" [von Neumann, 1958 p.131 trad.it. 2021].

Anche il sistema nervoso, allora, “quando lo si consideri come un automa, deve senza dubbio avere sia una parte aritmetica che una parte logica, e che in esso le esigenze dell’aritmetica sono altrettanto importanti di quelle della logica”. Diviene trattabile nuovamente, malgrado le differenze evidenziate sinora, come una macchina calcolatrice. Si pone subito una domanda, “considerando il sistema nervoso come una macchina calcolatrice, con quale precisione ci si può aspettare che funzioni la sua parte aritmetica?” [von Neumann, 1958 p.146 trad.it. 1994]

Si ripresentano le questioni della precisione, dell’errore e dell’affidabilità analizzate – e della differente gestione di questi – incontrate nella trattazione della teoria degli automi. Per affrontare questioni complesse come quelle trattate dal cervello, una “macchina deve essere dotata di dispositivi in grado di garantire livelli di precisione molto elevati” [von Neumann, 1958 p.133 trad.it. 2021].

Come abbiamo notato, nelle catene di calcolo di un computer, un errore, anche se molto piccolo, si diffonde con effetti catastrofici in termini di affidabilità. L’errore si accumula, si propaga e si amplifica nelle varie fasi del calcolo. “Ci si aspetta dunque che una componente aritmetica del sistema nervoso esista, e che se interpretata come macchina di calcolo, debba operare con notevole precisione”. Nell’ordine di complessità delle questioni trattate dal sistema nervoso, questo dovrebbe operare con una precisione di “dieci o dodici cifre decimali”. [von Neumann, 1958 p.133 trad.it. 2021].

Questa questione, per von Neumann, è decisiva “non tanto *nonostante* ma proprio *per* la sua assoluta irragionevolezza” [von Neumann, 1958 p.133 trad.it. 2021 enfasi originale]

Il sistema nervoso trasmette gli stimoli quantitativi come treni di impulsi periodici o quasi, la cui frequenza cresce monotonamente con l’intensità. Poiché le frequenze osservate variano solo tra 50 e 200 impulsi al secondo, la corrispondenza non è diretta con un valore numerico preciso. [Cfr. von Neumann, 1958 p.147 trad.it. 1994]. “Chiaramente, in queste condizioni, un grado di precisione come quello sopra menzionato (da 10 e 12 decimali!) è completamente fuori questione” [Cfr. von Neumann, 1958 p.147 trad.it. 1994].

Il cervello, l’automa rappresentato dal sistema nervoso, così come rappresentato da von Neumann, “non solo conduce a un *basso livello di precisione* ma anche a un livello piuttosto alto di *affidabilità*” [von Neumann, 1958 p.136 trad.it. 2021 enfasi mia].

In un computer, la mancanza di un singolo impulso digitale porta a una totale aberrazione del risultato della computazione. In un automa biologico un segnale incompleto, erroneo o mancante, varia la natura del messaggio in modo del tutto trascurabile.

Possiamo individuare il motivo di questa apparente contraddizione. Un calcolatore ha una “profondità logica” e una “profondità aritmetica” – ne sono esempi gli algoritmi *flow charts* con cui von Neumann e Goldstine programmarono le loro macchine – tali che un singolo errore produce conseguenze catastrofiche. “Il fatto che ci siano molte operazioni da eseguire in serie è, naturalmente, una caratteristica sia della struttura logica sia della struttura aritmetica di tale algoritmo. È quindi appropriato affermare che tutti questi fenomeni di deterioramento della precisione siano dovuti alla grande profondità logica degli algoritmi con cui si ha a che fare in questo contesto” [von Neumann, 1958 p.136 trad.it.].

Invece il sistema nervoso - e qui confluiscono le considerazioni di “riforma” in senso probabilistico della logica fatte da von Neumann trattando dell’inaffidabilità strutturale dei componenti degli automi – “ha carattere essenzialmente *statistico*” [von Neumann, 1958 p.136 trad.it. 2021 enfasi originale]. La notazione impiegata dal sistema nervoso non dipende, come nel sistema rigido di logica formale che von Neumann voleva integrale, dalla posizione – assenza/presenza – dei marcatori per determinare il significato di una trasmissione di informazione. Nell’automa biologico abbiamo a che fare “con un sistema di notazione in cui il significato del messaggio è veicolato dalle proprietà *statistiche* del messaggio” [von Neumann, 1958 p.137 trad.it. 2021 enfasi originale]. La minore precisione aritmetica è barattata con un maggior livello di profondità logica.

Nel caso funzionasse secondo i criteri e la precisione logica del calcolatore digitale – quelli del modello “classico” di McCulloch-Pitts – con un grado così basso di precisione come quello indicato, il cervello sarebbe “un asino computazionale” [Churchland, Churchland, 2000, p.38, trad. it. 2021]. Invece l’automa biologico non è né un asino né una tartaruga. “A prescindere dalla sua cautela riguardo ai dettagli del funzionamento del cervello, von Neumann sembra certo di una cosa: i cervelli calcolano” [Piccinini, 2003, trad. mia]⁷⁹⁵.

Nonostante le cautele sulla natura strutturale delle componenti degli automi biologici e delle modalità con cui eseguono le computazioni, la compensazione del basso grado di precisione si deve alla natura statistica dell’elaborazione delle informazioni insieme a un’organizzazione che si configura come parallela e su base sinaptica [Churchland, Churchland, 2000, pp.38-39, trad. it. 2021].

⁷⁹⁵ “But regardless of his caution about the details of brain function, von Neumann appears sure of one thing: brains compute”.

Dal confronto effettuato da von Neumann tra cervello e computer si potrebbe dedurre che una equiparazione tra i due sia improponibile⁷⁹⁶. L'integrazione della logica degli automi può servire a confutare una delle più celebri obiezioni all'interpretazione del cervello come elaboratore di informazioni, quella espressa nell'"assunto biologico" di Dreyfus: "Un assunto biologico secondo il quale, ad un certo livello di operazione – che di solito si suppone essere quello dei neuroni – il cervello tratta le informazioni in operazioni distinte attraverso un equivalente biologico di un interruttore posto o meno in funzione (on/off)" [Dreyfus, 1972, p. 235, trad. it.]. Al contrario, l'intera trattazione degli automi, culminata nello studio del più complesso di essi, il cervello, ha rappresentato per von Neumann il tentativo di mostrare come la rigidità della logica formale/digitale dovesse essere riformata: ampliandone la profondità, ammettendo la possibilità di malfunzionamenti con probabilità non nulla e orientandola, integrata con la teoria dell'informazione, persino verso una pseudomorfosi di tipo neurologico.

Le ultime considerazioni conducono alle riflessioni con cui si conclude *The Computer and the Brain*. Queste riguardano la natura del *linguaggio*: "Il sistema nervoso si basa su due tipologie di comunicazione: vi sono comunicazioni che non coinvolgono formalismi aritmetici e comunicazioni che li coinvolgono. Sono, nell'ordine, le comunicazioni (logiche) che trasmettono numeri. Le prime possono essere descritte come un linguaggio proprio, le seconde possono essere ricondotte alla matematica"

Il linguaggio è riconosciuto da von Neumann come un fatto "in gran parte un fatto storico accidentale". I linguaggi naturali umani, nella loro forma e nella loro molteplicità mostrano come non vi sia "un'unica forma necessaria di linguaggio". Parimenti a questi anche "i numeri e le costruzioni che li associano alla logica e alla matematica" è possibile che "siano un'espressione altamente storica e accidentale. Possono presentare varianti radicalmente differenti, possono esistere cioè in forme completamente diverse da quelle cui siamo abituati" [von Neumann, 1958, p.138-139 trad. it. 2021].

A dare la prova di questa *storicità* del linguaggio logico-matematico, è per von Neumann "è proprio la natura del sistema nervoso centrale e dei sistemi di messaggi che esso trasmette a indicarci che le cose stiano effettivamente così" [von Neumann, 1958, p.138 trad. it. 2021].

La profondità del linguaggio del sistema nervoso, qualunque esso impieghi, è minore a quella tradizionalmente considerata. Un esempio, per von Neumann, è quello della visione. La retina, riorganizza l'immagine visiva già al punto di ingresso del nervo ottico attraverso tre sole

⁷⁹⁶ Su come *The Computer and The Brain* dimostri, invece, l'impossibilità di questo confronto si veda Peters, 2018.

sinapsi consecutive, equivalenti a tre passaggi logici. Questo mostra come il sistema nervoso operi con processi a bassa profondità aritmetica e di natura statistica [Cfr. von Neumann, 1958, p.138 trad. it. 2021]

Si può leggere in queste parole di von Neumann una sorta di invito alla prudenza nei confronti di un eccessivo impegno ontologico nell'equiparazione tra cervelli e computer [Cfr. Israel, Millán Gasca, 2008, pp.228-229]. La cautela di prendere "*cum grano salis*" paragoni in senso forte tra sistema nervoso e i calcolatori come elaboratori digitali "rigidi" è espressa dallo stesso Shannon⁷⁹⁷, in una prospettiva simile a quella del matematico nato a Budapest, nel suo lavoro sugli automi:

La nostra conoscenza del funzionamento del cervello è ancora molto primitiva, nonostante una grande quantità di importanti e illuminanti ricerche. Ad esempio resta ancora aperta la questione se il livello neuronico sia il più adatto per un'analisi funzionale. La struttura casuale, al livello neuronico, del numero, della dislocazione e delle interconnessioni tra neuroni, suggerisce che solo gli aspetti statistici abbiano importanza a tale livello, e che, quindi, prima di costruire un modello matematico si potrebbe far la media sulla struttura e sul funzionamento locale [Shannon, 1958, pp.94-95 trad. it.].

A queste si possono connettere le analisi di von Neumann sul "carattere statistico del sistema di comunicazione utilizzato nell'aritmetica del sistema nervoso centrale e la sua bassa

⁷⁹⁷ Vale la pena sottolineare e riportare la precisa analisi fatta da Shannon, la cui formulazione dell'informazione è stata decisiva nella teoria degli automi di von Neumann, sulle analogie e differenze tra calcolatori e cervelli, in quanto fornisce un utile riepilogo dei temi qui trattati: "Le somiglianze tra cervello e calcolatori sono state spesso messe in rilievo. Le differenze sono forse più illuminanti, poiché possono suggerire importanti caratteristiche assenti dai nostri migliori attuali modelli del cervello. Tra le più importanti: 1) Differenze di dimensioni. Una differenza nel numero dei componenti di ben sei ordini di grandezza ci allontana tanto dalla nostra esperienza ordinaria da rendere quasi priva di senso ogni estrapolazione di funzioni. 2) Differenze di organizzazione strutturale. La struttura locale apparentemente casuale delle reti nervose è profondamente diversa dal preciso cablaggio degli automi artificiali in cui una sola connessione errata può essere causa di cattivo funzionamento. Il cervello invece è fatto in modo che il funzionamento complessivo non dipenda dall'esattezza della struttura nei particolari. 3) Differenze di affidabilità. Il cervello può operare per decenni senza casi di cattivo funzionamento veramente gravi (paragonabili alle espressioni prive di senso prodotte da un calcolatore in condizioni critiche) anche se individualmente i suoi componenti non sono più efficienti di quelli adoperati nei calcolatori. 4) Differenze di organizzazione logica. Qui le differenze sono tante che non possono essere enumerate. Il cervello ha la peculiarità di organizzarsi da sé in larga misura. Esso può adattarsi abbastanza bene a una vasta gamma di situazioni. Possiede notevoli caratteristiche di classificazione e di rapidità di accesso alla memoria, e la capacità di individuare rapidamente i dati immagazzinati mediante numerosi "sistemi di coordinate". Può allestire stabili servosistemi che richiedono complesse relazioni tra i suoi ingressi sensoriali e le sue uscite motorie, con grande facilità. Invece i nostri calcolatori numerici sono come degli *idiot savants*: persone incolte, ma capaci di fare mentalmente lunghi calcoli aritmetici. Finché si tratta di effettuare lunghe catene di operazioni numeriche essi si comportano molto meglio dei migliori calcolatori umani. Quando però si cerca di programmarli per altre attività, la situazione si rovescia: la programmazione appare grossolana e inadeguata. 5) Differenze di dispositivi di ingresso e uscita. Il cervello è dotato di organi di ingresso molto ben congegnati, specie l'orecchio e l'occhio, per la percezione dello stato del proprio ambiente". [Shannon, 1958, p.95 trad.it.].

precisione". Esse mostrano chiaramente che, "dovendo esistere qui strutture logiche diverse da quelle a cui siamo normalmente abituati e a cui facciamo riferimento in logica e in matematica, e come conseguenza inevitabile, devono esistere linguaggi caratterizzati da una minore profondità logica e aritmetica rispetto ad altre circostanze simili e più familiari" [von Neumann, 1958, p.138 trad. it. 2021]. Rispetto a una matematica e a una logica primarie del sistema nervoso centrale – sempre che esse esistano – i linguaggi formalizzati convenzionali "non sono che un'espressione secondaria" [von Neumann, 1958, p.138 trad. it. 2021].

Queste considerazioni non esulano però dalla *logica* del calcolatore (e del sistema nervoso) e dal modo con cui questo può ricevere istruzioni logiche, quello che si indica con il termine di "codice". "Per istruzioni logiche", spiega von Neumann, "io intendo, ad esempio, gli impulsi nervosi che compaiono su assoni appropriati e ogni altra cosa che induca un sistema logico digitale, come il sistema nervoso, a funzionare secondo modalità riproducibili per il raggiungimento di un fine determinato" [von Neumann, 1958, p.143 trad. it 1994].

Qui von Neumann, rifacendosi al lavoro di Turing, introduce una distinzione tra "*codice completo*" e "*codice breve*". Il primo consiste in una serie di istruzioni totalmente specificate, che definiscono, nel caso del sistema nervoso, l'ordine degli impulsi sugli assoni oppure, nel calcolatore, il comportamento di una macchina - ossia l'insieme di istruzioni con tutte le specificazioni necessarie per risolvere un problema computazionale. Come scrive von Neumann: "L'uso di un moderno calcolatore è basato sull'abilità di chi lo usa nello sviluppare e nel formulare i codici completi necessari per ogni singolo problema che la macchina deve risolvere" [von Neumann, 1958, p.144, trad. it. 1994]

Il secondo, il "codice breve", è un sistema di istruzioni che consente a una macchina di operare come se fosse un'altra, traducendo comandi complessi in operazioni elementari, sulla base del principio dimostrato da Turing nel 1936 in *On Computable Numbers*. Per questo deve includere ordini espressi in una forma che la macchina sia in grado di comprendere ed eseguire, così da poter riconoscere se un'istruzione ricevuta appartiene alla struttura propria della seconda macchina e, al tempo stesso, tradurla in comandi equivalenti formulati nel linguaggio della prima. In questo modo la prima macchina può effettivamente compiere le stesse azioni che la seconda avrebbe eseguito e, di conseguenza, riprodurre il comportamento in maniera completa [Cfr. von Neumann, 1958, pp. 144-145, trad. it. 1994]. Sottolinea von Neumann:

Il risultato importante raggiunto dallo schema di Turing è che in questo modo si riesce a far sì che la prima macchina imiti il comportamento di *qualsiasi* altra macchina. La struttura delle istruzioni

che vengono eseguite può essere completamente diversa da quella caratteristica della macchina che è realmente impegnata. In tal modo la struttura delle istruzioni può comprendere delle istruzioni di carattere molto più complesso di quelle 'caratteristiche della prima macchina: ognuna delle istruzioni della seconda macchina può richiedere l'esecuzione di molte operazioni da parte della prima [von Neumann, 1958, p.145, trad. it 1994 enfasi originale].

Queste considerazioni sulla codifica degli automi e sulle modalità di *imitazione* sembra autorizzare un'analogia forte tra il sistema di elaborazione biologico e quello artificiale. Al contrario, come mettono in evidenza Israel e Millán Gasca [Cfr. Israel, Millán Gasca, 2008, p.229], questa analisi va letta in relazione alla riflessione sulla natura storica del linguaggio della matematica rispetto a quello del sistema nervoso appena analizzate:

Occorre notare, inoltre, che il linguaggio di cui ci siamo occupati finora potrebbe corrispondere a un codice breve nel senso descritto in precedenza, e non a un codice completo: quando parliamo della matematica, potrebbe essere che stiamo discutendo di un linguaggio secondario, costruito sul linguaggio *primario* realmente utilizzato dal sistema nervoso centrale. In tal modo, le forme esteriori della *nostra* matematica non avranno assolutamente alcuna rilevanza dal punto di vista della valutazione di quale realmente è il linguaggio matematico o logico utilizzato dal sistema nervoso centrale [von Neumann, 1958, p.139-140 trad.it 2021].

In tal modo, pur paragonando il linguaggio del sistema nervoso a quello che in termini moderni corrisponde al "linguaggio macchina", (mentre l'altro corrisponderebbe a un "linguaggio di programmazione di alto livello") [Cfr. Churchland, Churchland, 2000, p.32 trad. it. 2021], von Neumann "non andava al di là di una prudente analogia formale fra cervello e calcolatore, aliena da qualsiasi pretesa ontologica di dire quale sia il linguaggio di cui «realmente» fa uso il cervello [Israel, Millán Gasca, 2008, p.229].

Ciò che invece von Neumann ribadiva era l'impiego dell'analisi matematica - come indicato nella missiva a Wiener del novembre 1946 - per trattare gli automi con lo stesso metodo rigoroso adottato da Turing. Ne è, a mio avviso, testimonianza l'osservazione conclusiva di *The Computer and the Brain*: "Tuttavia, le precedenti osservazioni circa l'affidabilità e la profondità logica e aritmetica dimostrano che, qualunque sia tale linguaggio, esso non può non differire notevolmente da ciò che consapevolmente ed esplicitamente pensiamo sia la matematica" [von Neumann, 1958, p. 140; trad. it. 2021].

Il livello di analisi matematica impiegato da von Neumann nello studio degli automi - dalla gestione della complessità dei meccanismi di organizzazione e autoriproduzione delle componenti elementari e dalla loro trattazione stocastica, sino all'analisi informazionale del sistema nervoso, che si colloca nella scia dell'elaborazione di metodi probabilistici comuni tanto alla cibernetica quanto al metodo Monte Carlo -, proprio perché disimpegnato dal livello fisico, non precludeva l'implementazione di automi artificiali secondo un'architettura seriale. Come sottolinea William Aspray, von Neumann "prevedeva che i progettisti di computer potessero trarre beneficio dal modellare nelle loro progettazioni alcune caratteristiche del sistema nervoso umano, ma metteva in guardia circa le possibili difficoltà" [Aspray, 1990, p. 209, trad. mia]⁷⁹⁸.

È possibile allora mettere in luce, a partire dalle riflessioni logiche, strutturali e organizzative emerse dalla trattazione del sistema nervoso - concepito come automa biologico -, una comune influenza tra la conoscenza del cervello e la costruzione dei computer. Tale nesso apre la via a un paradigma di design alternativo rispetto alla classica architettura di von Neumann: quello del connessionismo, già più volte richiamato in queste pagine, che offrirà uno spiraglio d'analisi sui recenti sviluppi delle tecnologie digitali e dell'Intelligenza Artificiale contemporanea.

⁷⁹⁸ "He anticipated that computer designers could profit by modeling features of the human nervous system in their designs but sounded a cautionary note on possible difficulties".

Cap.7: Connessioni: reti neurali, prospettive critiche, design

1. Dall'automa cerebrale alle reti: avventure e disavventure del Percettrone

Uno degli obiettivi del lavoro compiuto sugli automi da von Neumann era quello per cui l'implementazione di quelli artificiali si modellasse su quello del più complicato di quelli biologici, l'automa "sistema nervoso".

A partire da questa idea e dai guadagni concettuali ottenuti nella trattazione della teoria degli *automata* del matematico ungherese possiamo meglio definire i tratti di quella prospettiva, più volte incontrata in queste pagine, – anch'essa "legata all'analogia mente-computer" [Di Francesco, 2003, p.17] – che prende il nome di *connessionismo*. Questo approccio "alla modellizzazione della percezione e della cognizione che impiega esplicitamente alcuni dei meccanismi e degli stili di elaborazione ritenuti presenti nel cervello" [Barrow, 1996, p.135 trad. mia]⁷⁹⁹. Dalle analisi svolte è possibile delineare sorta di genealogia di un paradigma alternativo a quello del *design* di von Neumann che prende le mosse da proprio dall'invito dello stesso matematico ungherese allo studio rigoroso dell'automa "cervello", disvelando come questo abbia preso piede sino a diventare il retroterra dell'attuale diffusione pervasiva dell'Intelligenza Artificiale.

Rispetto alla classica architettura seriale – e nel suo analogo nella manipolazione simbolica della GOFAI – il connessionismo "propone una differente architettura del sistema: in particolare esso prende a modello la struttura biologica del cervello, offrendone un'elaborazione formale in termini di *reti neurali*" [Di Francesco, 2003, p.17 enfasi mia]. "I modelli di reti neurali sono stati ispirati e somigliano (in varia misura) all'anatomia e alla fisiologia del sistema nervoso. Un aspetto fondamentale di molti modelli di reti neurali è che essi sono in grado di apprendere e il loro comportamento può migliorare con l'addestramento o l'esperienza" [Barrow, 1996, p.135 trad. mia]⁸⁰⁰

L'analisi degli automi – nata nel cuore della cibernetica – ha mostrato la struttura *parallela* dell'elaborazione dell'informazione da parte del sistema nervoso. E parimenti, il

⁷⁹⁹ "Connectionism is an approach to modelling perception and cognition that explicitly employs some of the mechanisms and styles of processing believed to occur in the brain".

⁸⁰⁰ "In particular, connectionist models usually take the form of so-called neural networks, which are composed of a large number of very simple components wired together. Neural network models were inspired by and resemble (to varying degrees) the anatomy and physiology of the nervous system. A key aspect of many neural network models is that they are able to learn and their behavior may improve with training or experience".

connessionismo, prendendo come modello architetturale le reti neurali, di cui vedremo l'origine e la struttura. Queste sono intese come: "programmi "locali", che agiscono in parallelo, e (nelle varianti ortodosse) nega così l'esistenza di un livello intermedio autonomo di rappresentazioni, considerando piuttosto il mentale come emergente a partire dalla serie di "reti" che si auto-organizzano sulla base dell'interazione dell'individuo con l'ambiente" [Di Francesco, 2003, p.19]⁸⁰¹.

"L'idea che le capacità cognitive siano il risultato di connessioni o associazioni tra componenti ha una lunga storia che va da Aristotele fino agli empiristi britannici" [Buckner, Garson, 2019, p.77 trad. mia; Cfr. Medler, 1998, p.65]⁸⁰². Questa idea è poi proseguita nel lavoro di scienziati e psicologi ottocenteschi e di inizio Novecento come Herbert Spencer (1820–1903); William James (1842–1910) e Edward Thorndike (1874–1949) [Cfr Cordeschi, 1998, pp. 52-66; pp.77-81; Medler, 1998, pp.66-69].

In linea con le analisi sin qui svolte, si può assumere come punto di partenza il più volte citato modello di formalizzazione idealizzata del neurone, presentato come capace di computare le funzioni booleane, elaborato da McCulloch e Pitts in *A Logical Calculus* [McCulloch, Pitts, 1943]. Abbiamo altresì osservato come von Neumann abbia sottolineato, nello studio del sistema nervoso, la necessità di individuare un modello organizzativo diverso e più ampio rispetto al singolo neurone.

Sulla scia di questo è possibile individuare un altro momento fondativo nella storia del connessionismo, delle reti neurali e della computazione ispirata al sistema nervoso, nel lavoro dello psicologo candese, allievo di Karl Lashley, Donald O. Hebb (1904-1985). Nel suo *The organization of behavior* [Hebb, 1949], Hebb "formulò [...] una legge qualitativa sulla modifica delle connessioni nervose durante l'apprendimento, secondo la quale esse risultano rinforzate quando i relativi neuroni vengono simultanea-mente e ripetutamente eccitati" [Cordeschi, 1998, p.229]. Come nota C. Morabito, l'opera di Hebb segna "in qualche modo un'inversione di tendenza nella ricerca psicologica statunitense e il 'riemergere' [...] di quell'ipotesi associazionistica e neuroconnessionistica che [...] ha percorso come un fiume carsico lo sviluppo storico del pensiero psicologico fin dalle origini" [Morabito, 2006, p.87].

⁸⁰¹ Si legga in parallelo all'idea di autoorganizzazione quanto analizzato precedentemente rispetto agli automi cellulari (CA) di von Neumann.

⁸⁰² "The idea that cognitive abilities are the result of connections or associations between components has a long history going from Aristotle through the British Empiricists".

La teoria di Hebb era fondata, nei limiti delle conoscenze fisiologiche disponibili, su due tipi di evidenze a queste riducibili: 1) l'esistenza e le proprietà dell'attività cerebrale continua; 2) la natura della trasmissione sinaptica nel sistema nervoso centrale.

Combinando questi due principi, Hebb elaborò una teoria dell'apprendimento negli organismi, che - in coerenza con il paradigma "cibernetico" che sta attraversando il nostro lavoro - supera "la preclusione comportamentistica nei confronti della *black box*", della psicologia ed egli contemporanea e "reintroduce a pieno titolo nell'indagine psicologica la mente e il sistema nervoso" [Morabito, 2006, p.87].

Lo psicologo canadese sostenne che la stimolazione ripetuta di determinati recettori potesse, nel tempo, favorire la formazione di *cell assemblies* (assemblee cellulari) [Hebb, 1949, p.76], in grado di mantenere un'attività autonoma anche dopo la fine dello stimolo iniziale.

In questa prospettiva, l'attività cerebrale continua non solo garantiva lo spazio temporale necessario affinché avvengano modificazioni strutturali legate all'apprendimento, ma costituisce anche la forma più semplice di processo rappresentativo, fungendo da base per la nascita di immagini o idee [Cfr. Medler, 1998, p.70]⁸⁰³.

La connessione tra cellule nervose resta l'assunto di partenza, ma viene considerata la base fisica a partire dalla quale l'esperienza e l'apprendimento determinano il formarsi di 'sistemi' neuronali, all'interno dei quali l'attività di un neurone facilita l'attivazione di un altro come in un processo di risonanza. All'interno di un'assemblea cellulare, l'attivazione di un'unità innesca dunque un processo di eccitazione che si diffonde alle altre unità e 'ritorna', amplificato, sull'unità di partenza, e l'integrazione fra assemblee cellulari diverse (che Hebb chiama «sequenza di fase») è il fondamento fisico delle funzioni psichiche superiori. [Morabito, 2006, p.88]

Hebb formulò una legge, "il postulato di Hebb", che formalizzava l'apprendimento come modificazione delle connessioni sinaptiche tra neuroni:

Quando un assone della cellula A è abbastanza vicino da eccitare la cellula B, e prende ripetutamente parte alla sua eccitazione, hanno luogo, in una o in entrambe le cellule, dei processi di crescita o dei mutamenti strutturali tali da aumentare l'efficienza di A, in quanto cellula scatenante l'attività di B. [Hebb, 1948, p.129 trad. it. in Cordeschi, 1998, p.235]

⁸⁰³ Una prospettiva esaustiva del lavoro di Hebb nell'ambito più generale della storia della scienza cognitiva si trova in Boden, 2006, pp.268-281. Una chiara esposizione del pensiero di Hebb nella storia della neuropsicologia cognitiva è in Morabito, 2008, pp. 87-90.

A Hebb si deve il conio del termine “connessionismo” [Boden, 2006, p.270; Hebb, 1948, p.xix]. All’interno dell’alveo di questa prospettiva “l’apprendimento hebbiano è un algoritmo di addestramento non supervisionato, in cui la forza sinaptica (peso) viene aumentata se sia il neurone sorgente che quello bersaglio sono attivi nello stesso momento” [Medler, 1998, p.71 trad. mia]⁸⁰⁴. Una derivazione del principio di associazione è, in questa teoria, il suo corrispettivo inverso: il “*decadimento delle connessioni inutilizzate*” [Medler, 1998, p. 71; Hebb, 1940, pp. 229 e 232]. La forza della connessione sinaptica si indebolisce infatti quando i neuroni non risultano attivi simultaneamente. Dal lavoro di Hebb è possibile intravedere come il principio organizzativo per uno studio del cervello superi il singolo neurone verso dinamiche di connessione.

L’apprendimento hebbiano può essere formalizzato come:

$$w_{ij}(t+1) = w_{ij}(t) + net_i + net_j$$

dove $w_{ij}(t)$ è la forza sinaptica dal neurone i al neurone j al tempo t ; net_i è l’eccitazione del neurone sorgente e net_j quella del neurone destinatario dell’impulso [Medler, 1998 p.71].

Un altro momento cruciale della storia del connessionismo e delle reti neurali è stato il *Percettrone* [*Perceptron*] dello psicologo statunitense del *Cornell Aeronautical Laboratory* (CAL) Frank Rosenblatt (1928-1971) introdotto con l’articolo *The perceptron: A probabilistic model for information storage and organization in the brain* [Rosenblatt, 1958].

“Rosenblatt (1958) perfezionò i suggerimenti di Hebb formulandoli in modo completamente matematico. Egli si concentrò su semplici modelli di rete a due strati, che chiamò *percettroni*” [Buckner, Garson, 2019, p.77 trad. mia]⁸⁰⁵

L’intento di Rosenblatt era quello di “comprendere la capacità degli organismi superiori di riconoscimento percettivo, generalizzazione, richiamo e pensiero”. Per fare questo lo psicologo statunitense si poneva tre domande: “1) Come viene percepita, o rilevata, l’informazione sul mondo fisico dal sistema biologico? 2) In quale forma viene immagazzinata, o ricordata, l’informazione? 3) In che modo l’informazione contenuta nell’archivio, o nella

⁸⁰⁴ “Within connectionism Hebbian learning is an unsupervised training algorithm in which the synaptic strength weight is increased if both the source neuron and target neuron are active at the same time”.

⁸⁰⁵ “Rosenblatt (1958) refined Hebb’s suggestions by formulating them in full mathematical detail. He focused on simple two- layer network models, which he called perceptrons”.

memoria, influenza il riconoscimento e il comportamento?” [Rosenblatt, 1958, p. 386, trad. mia]⁸⁰⁶.

La prima domanda pertiene, secondo Rosenblatt, alla fisiologia, mentre egli intende dare risposta alla seconda e alla terza. “Rosenblatt respingeva esplicitamente l’approccio che chiamava «della logica simbolica, dei calcolatori digitali e della teoria dei circuiti», quello che egli faceva risalire ai pionieristici lavori di McCulloch e Pitts” [Cordeschi, 1998, p. 258]. Lo psicologo americano, invece, si rifaceva a quel “numero relativamente ridotto di teorici, come Ashby e von Neumann” che “si è invece interessato ai problemi di come una rete neurale imperfetta, contenente molte connessioni casuali, possa essere resa capace di svolgere in modo affidabile quelle funzioni che potrebbero essere rappresentate da schemi di cablaggio idealizzati”. Al pari di von Neumann, anche Rosenblatt denunciava come il “linguaggio della logica simbolica e dell’algebra booleana è meno adatto a tali indagini”. Anch’egli, per descrivere organizzazioni complessive delle quali la struttura è ignota, fu portato a “formulare l’attuale modello in termini di *teoria della probabilità* piuttosto che di logica simbolica” [Rosenblatt, 1958, p.388 trad. mia enfasi mia]⁸⁰⁷. In questo, Rosenblatt si riconnetteva al programma di ricerca cibernetico. [Boden, 2006, p. 905]⁸⁰⁸.

Alla serialità dell’elaborazione dell’informazione propria dell’architettura del calcolatore nell’approccio “classico” (tanto nell’IA quanto nelle scienze cognitive), Rosenblatt opponeva, in continuità con il connessionismo hebbiano, una prospettiva secondo cui “l’informazione [...] è contenuta in connessioni o associazioni piuttosto che in rappresentazioni topografiche” [Rosenblatt, 1958, p.386 trad. mia enfasi originale]⁸⁰⁹.

Per lo psicologo del CAL “nuova informazione equivale a «creazione» di nuove connessioni, sicché il riconoscimento o la classificazione degli stimoli (in effetti, la risposta appropriata) avviene tramite l’attivazione automatica di quelle connessioni” [Cordeschi, 1998, p.258].

⁸⁰⁶ “1. How is information about the physical world sensed, or detected, by the biological system? 2. In what form is information stored, or remembered? 3. How does information contained in storage, or in memory, influence recognition and behavior?”

⁸⁰⁷ “A relatively small number of theorists, like Ashby and von Neumann, have been concerned with the problems of how an imperfect neural network, containing many random connections, can be made to perform reliably those functions which might be represented by idealized wiring diagrams. Unfortunately, the language of symbolic logic and Boolean algebra is less well suited for such investigations. The need for a suitable language for the mathematical analysis of events in systems where only the gross organization can be characterized, and the precise structure is unknown, has led the author to formulate the current model in terms of probability theory rather than symbolic logic”.

⁸⁰⁸ “The theoretical language needed, Rosenblatt said, wasn’t Boolean logic but-as the cyberneticians had suggested-probability theory”.

⁸⁰⁹ “The information is contained in *connections* or *associations* than topographic representations”.

A tal fine Rosenblatt, realizzò la sua rete neurale, il *Percettrone* appunto, sul modello dei sistemi biologici [Cfr. Rosenblatt, 1958, p.387]⁸¹⁰. Questo [fig.26] è un modello artificiale composto da tre livelli: uno strato sensoriale (*S-unit*), che riceve stimoli dall'ambiente e li traduce in segnali; uno strato intermedio di unità associative (*A-unit*), che elaborano le informazioni; e uno strato di risposta (*R-unit*), che produce l'output verso l'esterno. [Cfr. Rosenblatt, 1958, pp.389-393].

Le connessioni tra i neuroni degli strati di input e di output sono modellate come pesi sinaptici v_i che possono assumere valori positivi o negativi. Questi pesi vengono moltiplicati per l'attività del neurone di input $u_i(t)$ per rappresentare il grado con cui la sua attivazione contribuisce all'attivazione, o alla mancata attivazione, del neurone di destinazione. La funzione di trasferimento tra due unità è quindi: $c_{ij}(t) = u_i(t) \cdot v_{ij}(t)$. Le connessioni tra input e unità associative rimangono fisse, mentre quelle tra unità associative e di risposta possono variare secondo una regola di rinforzo, descritta da: $v_{ij}(t + 1) = v_{ij}(t) + \Delta v_{ij}$ (con $\Delta v_{ij} = u_i(t)$ se l'unità i è attiva al tempo t , altrimenti 0) [Cfr. Medler, 1998, p.73].

In questo modo, il percettrone apprende a classificare i pattern binari rafforzando o indebolendo le connessioni a seconda della *storia* di apprendimento. Come scrive la computer scientist, sintetizzando la matematizzazione della rete neurale di Rosenblatt, Melanie Mitchell:

Analogo al neurone, il percettrone somma gli input che lo raggiungono, e se la somma risultante è uguale o maggiore della soglia del percettrone, quest'ultimo restituisce come output il valore 1 (scarica); altrimenti restituisce il valore zero (non scarica). Per simulare le differenti forze di connessione su un neurone, Rosenblatt propose di assegnare un peso numerico a ciascun input del percettrone; ciascun input è moltiplicato per il proprio peso prima di essere aggiunto alla somma. La soglia di un percettrone è semplicemente un numero stabilito dal programmatore (o [...] appreso dal percettrone stesso) [...] In breve, un percettrone è un programma semplice che prende una decisione sì-o-no (1 o zero), basata sul fatto che la somma dei suoi input ponderati corrisponda a un valore soglia [Mitchell, 2019, pp.13-14 trad.it].

⁸¹⁰ "The theory has been developed for a hypothetical nervous system, or machine, called a perceptron. The perceptron is designed to illustrate some of the fundamental properties of intelligent systems in general, without becoming too deeply enmeshed in the special, and frequently unknown, conditions which hold for particular biological organisms. The analogy between the perceptron and biological systems should be readily apparent to the reader".

L'apparente contraddizione tra il meccanismo digitale "all-or-none" e la teoria probabilistica si supera osservando che, per Rosenblatt, le regole locali (soglia, pesi, rinforzo) sono deterministiche, mentre il comportamento complessivo della rete, data la molteplicità e casualità delle connessioni, è descrivibile solo in termini probabilistici.

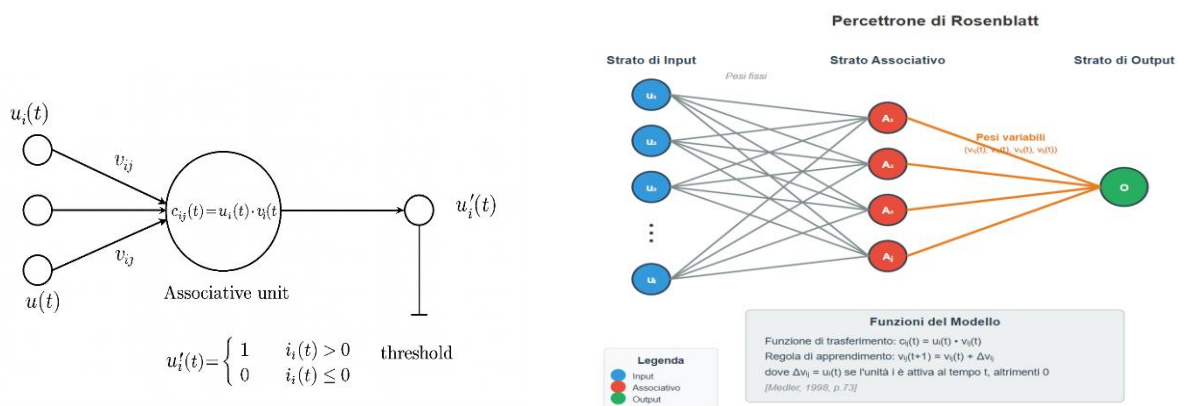


Fig. 26 lo schema del Percettrone di Rosenblatt

Il lavoro di Rosenblatt non rimase solamente quello di ideare una macchina astratta. Al CAL egli avviò, dal 1957, un programma di simulazioni digitali sul calcolatore IBM 704, finalizzate a testare le capacità del percettrone in compiti di apprendimento e riconoscimento di forme visive. Le matrici di stimoli erano tradotte in punti luminosi su una "retina" artificiale e trasformate in input per una rete simulata di unità associative e di risposta. In queste condizioni era possibile osservare processi di classificazione, generalizzazione e apprendimento sia in modalità di apprendimento forzato, in cui il sistema riceveva come rinforzo la risposta imposta dall'esperimento, sia in modalità di apprendimento spontaneo, in cui veniva rinforzata la risposta che il percettrone produceva autonomamente. Rosenblatt mise in evidenza anche le capacità del sistema di auto-organizzazione, cioè la possibilità che, esposto a una serie di stimoli, il percettrone arrivasse a formare spontaneamente classificazioni significative [Rosenblatt, 1960].

Parallelamente, il CAL realizzò una implementazione fisica, il *Mark I Perceptron*, che riproduceva in hardware la struttura teorica descritta da Rosenblatt nel 1958. Anch'esso era organizzato in tre livelli - *S-unit*, *A-unit* e *R-unit* - collegati da connessioni eccitatorie e inibitorie, con un meccanismo fisico di correzione degli errori. Il sistema era alimentato da dispositivi elettromeccanici e potenziometri regolabili, che permettevano di variare i pesi

sinaptici e simulare l'apprendimento. Il Mark I Implementava un sistema di retina artificiale, concepito per imitare la retina biologica: un mosaico di fotocellule disposte in griglia, capace di reagire alla luce e trasformare gli stimoli visivi in segnali elettrici binari. Questi venivano poi distribuiti in modo parzialmente casuale alle unità associative, che ne elaboravano l'attività determinando l'attivazione o meno delle unità di risposta. In tal modo il Mark I era in grado di tradurre configurazioni di punti luminosi in schemi interni di attivazione, dando vita a una forma rudimentale di "visione" artificiale, analoga al modello teorico [Kanal, 1962]⁸¹¹.

I progetti sul percettrone di Rosenblatt erano stati – in un ulteriore esempio di implementazione "sociotecnica" delle tecnologie digitali e dell'informazione – finanziati dall'Office of Naval Research (ONR), che ne promosse, con toni probabilmente troppo ottimisti, la diffusione con una celebre conferenza stampa a Washington, il 7 luglio 1958 [Olarazan, 1996, p.621]⁸¹². L'evento fu coperto dal *New York Times* che il 13 luglio scriveva:

La marina militare ha rivelato oggi l'embrione di un calcolatore elettronico che, a suo giudizio, sarà capace di camminare, parlare, vedere, scrivere, riprodursi ed essere cosciente della propria esistenza. E si prevede anche che, un domani, i percettroni sapranno riconoscere le persone, pronunciare ad alta voce il loro nome e tradurre all'istante la parola di una lingua nella parola e nella scrittura di un'altra lingua [New York Times, 13 luglio 1958 trad. it in Mitchell, 2019 trad. it.].

"Secondo la storia ufficiale della controversia, le "affermazioni eccessive" di Rosenblatt irritarono molte persone nella comunità dell'IA, inclusi alcuni dei suoi leader. I ricercatori di oggi ricordano che Rosenblatt era solito rilasciare dichiarazioni continue ed esagerate sulle prestazioni della sua macchina" [Olarazan, 1996, p.621 trad. mia]⁸¹³. Tra questi leader della GOFAI, infastidisti sia dal clamore delle dichiarazioni di Rosenblatt, e probabilmente dall'approccio differente che il modello esemplificato dal percettrone rappresentava, vi erano Marvin Minsky (1927-2016) e Seymour Papert (1928-2016).

Il perceptrone, come lo stesso Rosenblatt riconosceva nei suoi scritti, presentava infatti limiti concreti specialmente nei problemi di riconoscimento visuale: non riusciva a rilevare adeguatamente somiglianze complesse tra figure, classificava gli stimoli solo in base

⁸¹¹ Si rimanda allo stesso per lo schema dell'hardware del Mark I Perceptron.

⁸¹² Allo stesso si rimanda per una puntuale ricostruzione della polemica che riguardò il *Perceptron* di Rosenblatt opposto alla comunità dell'IA.

⁸¹³ "According to the official history of the controversy, Rosenblatt's 'overclaims' irritated many people in the AI community, including some of its leaders. Present day researchers remember that Rosenblatt was given to steady and extravagant statements about the performance of his machine".

all'“overlap” sulla retina di input, ed era privo di un vero sistema di *pre-processing* capace di distinguere le componenti di un'immagine e le relazioni tra esse. Questi vincoli riducevano la sua capacità di trattare problemi di riconoscimento più generali [Cfr. Olarazan, 1996, pp.623-624].

Minsky e Papert portarono queste osservazioni sui limiti del modello di Rosenblatt a un livello teorico sistematico, pubblicando *Perceptrons: An Introduction to Computational Geometry* [Minsky, Papert, 1969], un testo che ebbe un impatto decisivo sugli sviluppi successivi dell'IA. Il loro approccio non si limitava a chiedere se le reti neurali fossero “buone” o “cattive”, ma piuttosto per cosa fossero effettivamente buone. [Minsky, Papert, 1969, p.163]. Dimostrarono che i percettroni semplici erano limitati alle funzioni dette *linearmente separabili*, cioè a quei problemi in cui le classi da distinguere possono essere separate da un iperpiano nello spazio degli input. Un esempio emblematico di questo limite in un compito elementare era la funzione esclusiva XOR, semplice ma non linearmente separabile. [Medler, 1998, p.76].

Per risolvere questi problemi computazionali – che riguardavano anche la “filosofia di fondo” dell'approccio ai problemi dell'IA, contrapposto al rappresentazionalismo della GOF AI – erano necessarie reti neurali a più strati (multi-layer). Solo con questa architettura era possibile superare i limiti dei percettroni a strato singolo e affrontare compiti non linearmente separabili, come la funzione XOR. Rosenblatt aveva già intuito la centralità delle reti multistrato, pur riconoscendo le difficoltà legate al loro addestramento [Rosenblatt, 1962, pp. 577-580]. Poiché all'epoca non esistevano metodi efficaci per l'addestramento degli strati nascosti, i percettroni rimasero confinati ai problemi linearmente separabili [Medler, 1998, p. 76].

La critica contenuta in *Perceptrons* aveva un retrogusto “passivo-aggressivo” riguardo a questo sviluppo delle reti multi-layer [Mitchell, 2019, p.21]. Infatti i due autori scrivevano:

Il problema dell'estensione non è soltanto tecnico, ma anche strategico. Il percettrone si dimostrato degno di studio nonostante (e persino a causa di!) le sue gravi limitazioni. Ha molte caratteristiche che attirano l'attenzione: la sua linearità; il suo affascinante apprendimento; la sua chiara semplicità paradigmatica come una sorta di computazione parallela. Non vi è motivo di supporre che qualcosa di queste virtù si trasferisca alla versione a più strati. Tuttavia, consideriamo importante come problema di ricerca chiarire (o rifiutare) il nostro giudizio iniziale che l'estensione sia sterile. Forse verrà scoperto qualche potente teorema di convergenza, o si troverà una riprova empirica per il fallimento

nel produrre un “teorema di apprendimento” interessante per la macchina a più strati [Minsky, Papert, 1969, pp. 231-232 trad. mia]⁸¹⁴.

Queste valutazioni negative contribuirono, sul finire degli anni Sessanta, al progressivo esaurirsi dei finanziamenti destinati alle ricerche sulle reti neurali, mentre l’IA simbolica riceveva ingenti risorse governative. Nel 1971 Frank Rosenblatt morì prematuramente, a soli quarantun anni, in un incidente in barca. Venute meno la sua figura di riferimento e il sostegno economico, le ricerche sui percettroni e, più in generale, sul programma di ricerca inaugurato dal percettrone si arrestarono quasi del tutto, sopravvivendo solo in pochi gruppi accademici isolati, portando al primo “inverno”⁸¹⁵ dell’IA [Mitchell, 2019, p.21 trad.it.].

È utile dare un breve sguardo ai tentativi di elaborazione di modelli teorici e dei tentativi di implementazione in hardware delle idee di cui è stato emblema il *Perceptron* di Rosenblatt (e la sua ispirazione hebbiana).

Nel settembre del 1956, in occasione del *MIT Symposium on Information Theory* – evento ricordato come la nascita ufficiale delle scienze cognitive – fu presentato il lavoro di Nathaniel Rochester, protagonista anche dell’altra grande fondazione dello stesso anno, quella dell’intelligenza artificiale alla conferenza di Dartmouth. Insieme a John Holland (in seguito ideatore degli algoritmi genetici), L. H. Haibt e William L. Duda, Rochester condusse uno dei primi esperimenti di simulazione di reti neurali artificiali su un grande calcolatore, l’IBM 704. L’intento era verificare la teoria delle *cell assemblies* proposta da Donald O. Hebb, secondo cui l’apprendimento avviene attraverso il rafforzamento delle connessioni tra neuroni che si attivano insieme.

Nelle simulazioni, dapprima con 69 neuroni e poi con 512, ogni unità aveva variabili come soglia, fatica e peso sinaptico, regolati secondo una regola hebbiana di rinforzo e decadimento. I risultati furono significativi: nel modello a 69 neuroni si osservò la

⁸¹⁴ “The problem of extension is not merely technical. It is also strategic. The perceptron has shown itself worthy of study despite (and even because of!) its severe limitations. It has many features to attract attention: its linearity; its intriguing learning theorem; its clear paradigmatic simplicity as a kind of parallel computation. There is no reason to suppose that any of these virtues carry over to the many-layered version. Nevertheless, we consider it to be an important research problem to elucidate (or reject) our intuitive judgment that the extension is sterile. Perhaps some powerful convergence theorem will be discovered, or some profound reason for the failure to produce an interesting “learning theorem” for the multilayered machine will be found”.

⁸¹⁵ Nella storia dell’intelligenza artificiale si parla spesso di stagioni per descrivere l’andamento ciclico di entusiasmi e crisi. Con l’espressione *AI Winter* si indicano i periodi di sfiducia e di drastico calo dei finanziamenti, in cui la ricerca sull’IA viene di fatto “congelata” a causa di aspettative disattese e promesse non mantenute. Al contrario, i termini *AI Spring* o *AI Summer* vengono usati per descrivere le fasi di entusiasmo e di boom, quando i progressi tecnologici e i successi applicativi portano a un forte aumento di interesse e di investimenti.

reverberazione diffusa, proposta come meccanismo plausibile della memoria a breve termine, ma non emerse alcuna tendenza alla formazione spontanea di *cell assemblies*. Con il modello a 512 neuroni, invece, comparvero veri e propri assemblaggi di cellule, che mostravano fenomeni di organizzazione e ristrutturazione dinamica.

Rimaneva tuttavia un limite: le reti risultavano ancora troppo dipendenti dagli stimoli esterni, incapaci di avviare da sole nuove sequenze interne. In altri termini, non riuscivano a mostrare quell'attivazione autonoma e ricorsiva tra gruppi di neuroni che sarebbe stata necessaria per simulare processi cognitivi più complessi, come il pensiero. Questo segnò uno dei primi tentativi concreti di verificare, con la potenza disponibile in un calcolatore elettronico, ipotesi neuropsicologiche sull'organizzazione della memoria e dell'apprendimento. Il paradigma hebbiano veniva così messo alla prova su un computer, impiegato per sondarne limiti e potenzialità rispetto all'organizzazione delle reti neurali artificiali e alle regole di apprendimento per associazione [Rochester et. al, 1956, pp.80-93; Anderson, Rosenfeld, 1988, pp. 65-67].

Nel 1959 Oliver Selfridge (1926-2008), ricercatore al MIT e ai *Lincoln Laboratories* propose *Pandemonium*, un modello teorico di apprendimento per il riconoscimento di pattern non specificabili in anticipo. L'architettura era composta da più strati di "demoni": i *data demons* che trasmettono l'informazione, i *computational demons* che eseguono funzioni elementari, i *cognitive demons* che valutano l'evidenza e infine il *decision demon* che sceglie l'interpretazione più forte. Il sistema si adattava tramite due meccanismi: la pesatura delle caratteristiche con procedura di *hill-climbing*, un metodo iterativo che modifica gradualmente i pesi alla ricerca di un miglioramento, come un cieco che sale verso la cima seguendo i piccoli rialzi del terreno, e la selezione dei sotto-demoni, che eliminava quelli inefficaci e generava nuove combinazioni. Applicato a compiti concreti, come la distinzione fra punti e linee nel codice Morse, *Pandemonium* mostrò che i principi connessionisti potevano fornire una base operativa per la classificazione automatica, anticipando molte idee che diventeranno centrali nelle reti neurali successive [Selfridge, 1959; Cfr. Medler, 1998, pp.72-73].

Un ulteriore passo in avanti è stato quello di Bernard Widrow (1929-) e da uno dei futuri padri del microprocessore, Marcian Edward "Ted" Hoff (1937-), che nel 1960 presentarono l'Adaline (*ADaptive LINear NEuron*, poi *ADaptive LINear Element*) come evoluzione del perceptrone. Il modello, ispirato sia al neurone logico di von Neumann in *Probabilistic Logic and Synthesis of Reliable Organisms from Unreliable Components* sia alla matematica probabilistica di Wiener nello *Yellow Peril* [Wiener, 1949] [Widrow, Hoff 1960, in Anderson, Rosenfeld, p.129

e p.133], fu concepito come macchina adattiva per la classificazione di pattern. L'unità Adaline calcolava una combinazione lineare dei segnali in ingresso, producendo un output +1 o -1, e ne aggiornava i pesi con la cosiddetta *Widrow-Hoff rule* o *delta rule* [Anderson, Rosenfeld, 1988, p.124], che costituisce un'applicazione esplicita della tecnica del *gradient descent* per minimizzare l'errore quadratico medio tra uscita e valore atteso. "In breve, essa utilizzava la differenza esistente tra il pattern di stimoli dato in ingresso e quello cercato per modificare i pesi delle connessioni durante la fase di "addestramento" [Cordeschi, 1998, p.232].

Matematicamente la Widrow-Hoff rule si può rappresentare come:

$r_i = z[(t) - y(t)] x_i(t)$, dove t è il pattern obiettivo, $y(t)$ è l'output della rete e $x_i(t)$ è l'input alla rete. [Medler, 1998, p.75]

Per quanto l'apprendimento, riguarda gli autori parlano di un *boss* (o *teacher*), che conosce la risposta corretta e guida l'unità, e di un *worker*, che esegue e corregge i propri pesi finché l'errore non è ridotto a zero. Si configurava cioè una forma di apprendimento *supervisionato*. Il perceptrone modificava i pesi soltanto in caso di classificazioni errate, l'Adaline, invece, continuava ad apprendere anche quando la risposta era formalmente corretta ma non esattamente uguale a +1 o -1: questo garantiva un apprendimento più rapido, continuo e stabile.

Non a caso, la backpropagation, che come vedremo a breve segnerà la rinascita delle reti neurali negli anni '80, può essere considerata una generalizzazione della delta rule. [Widrow, Hoff 1960, pp.126-134; Cfr. Medler, 1998, p.74-75; Boden, 2006, pp.909-911]⁸¹⁶.

A differenza del perceptrone, che modificava i pesi solo in caso di errore di classificazione grossolano, l'Adaline continuava ad apprendere anche quando la risposta era corretta ma non esattamente uguale al valore atteso. Questo garantiva apprendimento continuo, più rapido e stabile, rappresentando un passo avanti decisivo verso l'uso sistematico dell'apprendimento supervisionato.

Malgrado il lavoro di Minsky e Papert avesse inferto un duro colpo alla ricerca sulle reti neurali, sotto la coltre di neve dell'"inverno" del connessionismo continuava a bruciare una fioca fiamma di ricerche pionieristiche nel campo.

James A. Anderson (1940-) sviluppò modelli di memoria associativa [Anderson, 1972], mostrando come reti distribuite potessero immagazzinare e recuperare pattern anche in presenza di errori o dati incompleti: un'anticipazione delle moderne memorie auto-associative.

⁸¹⁶ Lo schema di Adaline è in Medler, 1998, p.75.

Teuvo Kohonen (1934–2021), nello stesso anno, elaborò principi di auto-organizzazione nelle reti, culminati poi nelle *Self-Organizing Maps*, che dimostravano come un sistema neurale potesse strutturare autonomamente rappresentazioni spaziali e topologiche dei dati [Kohonen, 1972]. Stephen Grossberg (1939–2022) propose teorie matematiche generali sull'apprendimento adattivo (1976), introducendo i modelli di *Adaptive Resonance Theory* (ART), una teoria dell'elaborazione dell'informazione umana via via evolutasi in una serie di modelli di reti neurali in tempo reale capaci di spiegare il bilanciamento tra plasticità e stabilità nei processi di apprendimento e riconoscimento [Grossberg, 1976]. I modelli di Kohonen e Grossberg rappresentano tipi di *apprendimento non supervisionato* [Cfr. Pessa, 2001, pp.107-108; Medler, 1998, pp.78-85].

Negli stessi anni, nell'Unione Sovietica – dove la cibernetica aveva smesso di essere etichettata come “scienza reazionaria e borghese” [Gerovitch, 2002] – Alexey Ivakhnenko (1913–2007) e Valentin Lapa (1928–2004) svilupparono il *Group Method of Data Handling* (GMDH) [Ivanenkho, Lapa, 1966], primo vero tentativo di addestrare reti multilayer. Il loro metodo consisteva nel determinare e ottimizzare i pesi delle connessioni strato per strato, usando regressioni polinomiali e criteri statistici di selezione, così da costruire modelli progressivamente più complessi e accurati. Poco dopo, nel 1967, in Giappone, Shun-Ichi Amari (1936–) formulò un modello di percettrone *multilayer* apprendibile tramite *stochastic gradient descent* risolvendo per la prima volta in modo operativo il problema della *non-linear separability* [Amari, 1967].

Questi contributi costituirono le basi silenziose ma decisive per la “rivoluzione connessionista” degli anni '80. [Buckner, Garson, 2019, p.78].

Sul retroterra storico che abbiamo tracciato, e degli snodi concettuali che si sono via via presentati, è ora possibile illustrare meglio il funzionamento di una *rete neurale multistrato* [Fig. 27] le cui potenzialità presunte erano state il punto critico dell'attacco di Minsky e Papert al modello del percettrone.

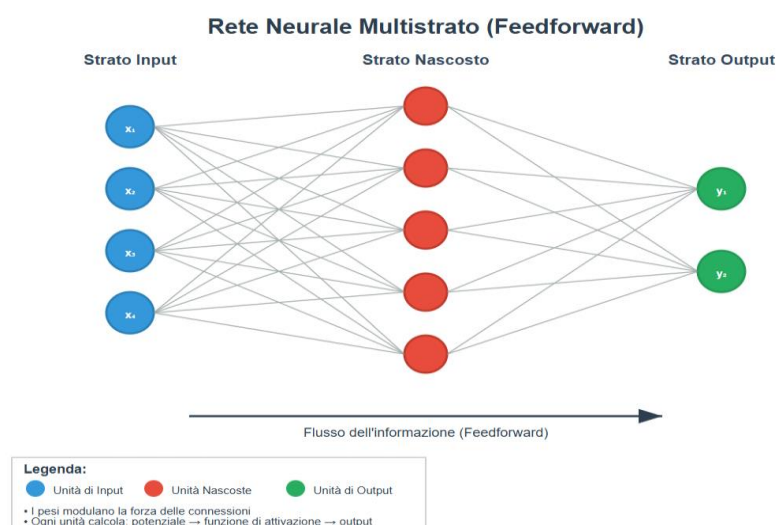
Una rete di questo tipo è composta da unità (o nodi), ciascuna delle quali funziona come un dispositivo ingresso-uscita: riceve segnali in ingresso, calcola un valore interno detto potenziale di attivazione, e restituisce in uscita un valore di attivazione. Questo valore dipende sia dagli ingressi ricevuti, sia dai *pesi* che modulano la forza o l'inibizione di ciascuna connessione. In generale, il potenziale è dato dalla somma pesata degli ingressi, e viene trasformato da una funzione di attivazione. Storicamente sono state usate funzioni diverse: lineare, a soglia, oppure sigmoide, che restituisce valori compresi tra 0 e 1.

La rete multistrato si distingue dal perceptrone per la presenza di uno o più strati nascosti [*hidden layers*], collocati tra gli strati di input e di output. Questi strati rappresentano la vera novità rispetto al modello a un *layer* solo che abbiamo incontrato nel perceptrone nel quale l'unità si limita a decidere se "scaricare" o no in base a una soglia, le unità nascoste possono trasformare stimoli semplici in rappresentazioni intermedie più complesse. Così, da un insieme di input elementari, come i pixel di un'immagine, la rete può arrivare a estrarre forme ricorrenti, configurazioni o tratti caratteristici, che diventano la base per la classificazione a livello di output.

L'aggiunta di strati nascosti rappresenta un primo passo per il superamento delle critiche mosse da Minsky e Papert. "Per aumentare la potenza computazionale dei perceptroni sarebbe stato sufficiente aggiungere uno o più livelli di unità nascoste fra lo strato di unità di input e lo strato di unità di output [...]. Tuttavia, per reti di questo genere la semplice versione della regola *delta* [...] non è più utilizzabile, in quanto essa non dice nulla su come aggiornare il peso di connessioni che collegano direttamente unità di input con unità di output" [Frixione, 1992, p.32]

Un'architettura tipica di rete neurale è la rete *feedforward*, in cui le connessioni vanno sempre dallo strato di input a quello di output attraverso i livelli nascosti, senza retroazioni. Dal punto di vista dell'informazione, ciò che conta è che questa non è localizzata in un singolo nodo, bensì è distribuita nella struttura dei pesi dell'intera rete: sono i valori delle connessioni a determinare la capacità del sistema di riconoscere e generalizzare [Buckner, Garson 2019b: Mitchell, 2019; pp.24-27; Pessa, 2001, pp.99-100].

Fig. 28



A segnare la rinascita del paradigma fondato sulle reti neurali hanno giocato almeno due fattori. In primo luogo il diffuso malcontento per i risultati ottenuti dalla ricerca all'interno del paradigma classico. I successi iniziali degli anni Sessanta avevano infatti alimentato nei protagonisti della cosiddetta GOFAI aspettative del tutto irrealistiche: si riteneva che nel giro di uno o due decenni i computer sarebbero stati in grado di eguagliare l'intelligenza umana. Alla fine degli anni Settanta, però, era ormai chiaro che quegli obiettivi non erano stati raggiunti e che il loro mancato conseguimento non aveva nemmeno fornito nuove indicazioni utili per il futuro. Il terreno era quindi pronto per l'affermarsi di un approccio alternativo [Buckner, Garson, 2019, p.78].

Il secondo è stato l'introduzione di un algoritmo migliorativo per l'apprendimento delle reti neurali che "generalizza la regola delta", capace una volta "confrontato l'output ottenuto con quello atteso", di rendere possibile la "propagazione all'indietro" [...] dell'errore mediante la modifica dei pesi delle connessioni interne. L'individuazione di questa "regola delta generalizzata" fu uno dei fattori determinanti per la rinascita dell'interesse delle reti neurali" [Frixione 1992, pp.32-33].

"L'accostamento di Papert tra i *Perceptrons* e il cacciatore mandato a riportare il cuore di Biancaneve è calzante: infatti il cacciatore non tornò con il cuore di Biancaneve, ma con quello di un cervo. Allo stesso modo, il connessionismo non fu ucciso dai *Perceptrons*; rimase semplicemente in attesa del proprio momento, fino all'arrivo del suo principe. E per il connessionismo, il Principe Azzurro si rivelò essere la Regola Delta Generalizzata (GDR)" [Medler, 1998, p.87 trad. mia]⁸¹⁷.

2. Un principe azzurro subsimbolico. Backpropagation, parallelismo e macchine probabilistiche

Il nome con cui questo Principe computazionale è diventato noto è quello di *backpropagation*. La retropropagazione dell'errore è un algoritmo che funziona calcolando innanzitutto l'errore allo strato di output (O), cioè la differenza tra l'attività effettiva di un neurone e quella desiderata. Successivamente vengono esaminati i neuroni nello strato nascosto (H) che proiettano verso i neuroni di output, e per ciascuno di essi si determina un

⁸¹⁷ "Papert's likening of Perceptrons to the huntsman being sent out to bring back Snow Whites heart is appropriate for the huntsman did not return with the heart of Snow White but the heart of a deer Similarly connectionism was not slain by Perceptrons it was just quietly minding its time until its prince came And for connectionism Prince Charming turned out to be the Generalized Delta Rule (GDR)".

errore in base al grado con cui la sua attivazione contribuisce a causare (o a correggere) errori nelle unità di output. I pesi tra lo strato nascosto e quello di output vengono quindi modificati leggermente nella direzione che riduce l'errore. Lo stesso procedimento si applica via via agli strati precedenti, fino a risalire allo strato di input [Cfr. Buckner, Carson, 2019, p.78]⁸¹⁸

Se immaginiamo un paesaggio in uno spazio con una dimensione per ciascun peso e in cui l'altezza del terreno rappresenta l'errore per un dato insieme di valori dei pesi, allora la backpropagation può essere vista come un metodo per attraversare questo spazio nella direzione che porta al punto più basso (ossia il minimo errore). Un segreto del successo della backpropagation era l'uso di valori graduati sia per l'attività dei neuroni sia per i pesi, che rendevano più levigato il paesaggio degli errori, cosicché la ricerca di un minimo aveva maggiori probabilità di riuscire [Buckner, Carson, 2019, p.79 trad. mia]⁸¹⁹.

L'idea di calcolare le derivate "all'indietro" fu formulata per la prima volta da Paul Werbos (1947–2021) nella sua tesi di dottorato *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences* (Werbos, 1974). Werbos propose la retropropagazione come metodo generale per ottimizzare sistemi complessi, anticipando di oltre un decennio la sua applicazione sistematica alle reti neurali. Risultati analoghi erano stati raggiunti nel 1985 da Y. LeCun (LeCun, 1985). La svolta decisiva si ebbe nel 1986, quando David E. Rumelhart (1942–2011), Geoffrey E. Hinton (1947–) e Ronald J. Williams (1953–) ripresero e svilupparono la retropropagazione, pubblicando due lavori fondamentali nel 1986, *Learning representations by back-propagating errors* su *Nature* [Rumelhardt, et. al, 1986a] e *Learning internal representations by error propagation* [Rumelhardt et al. 1986b].

Rumelhart, Hinton e Williams dimostrarono che reti multistrato addestrate con la *backpropagation* erano in grado di risolvere problemi che i percettroni a singolo livello non potevano affrontare, come la funzione XOR. Gli articoli mostravano, inoltre, che i livelli nascosti sviluppavano spontaneamente *rappresentazioni* interne, ossia codici distribuiti che ricodificavano gli input in forme più adatte alla classificazione e al riconoscimento [Rumelhardt et. al, 1986a; Rumelhardt et al. 1986b].

⁸¹⁸ Si rinvia a Medler, 1998, pp.87-89 per la formalizzazione matematica della GDR/backpropagation. Mitchell, 2019, pp.26-28 ne fornisce invece una descrizione accessibile al lettore non specialista.

⁸¹⁹ If we imagine a landscape in a space with a dimension for each weight and where the height of the terrain is the error for a given set of weight values, then backpropagation can be seen as a method for traversing the space in the direction that will find the lowest point (lowest error). A secret to the success of backpropagation was the use of graded values for both the activity of neurons and the weights, which smoothed the error landscape so that a search for a minimum was more likely to succeed.

I risultati segnarono un vero punto di svolta: la retropropagazione si affermò come il primo metodo realmente praticabile per l'addestramento di reti multistrato. Il suo valore stava nella generalità, perché poteva essere applicata a qualunque rete feedforward con funzioni di attivazione derivabili; nell'efficienza, che consentiva di gestire reti con un gran numero di pesi; e infine nella potenza rappresentazionale, poiché rendeva possibile apprendere strutture distribuite e complesse all'interno degli strati nascosti.

Sebbene la retropropagazione di base sia abbastanza potente da risolvere una grande varietà di problemi, molti lavori si concentrano sul miglioramento delle prestazioni delle reti neurali artificiali, in particolare rispetto a tre caratteristiche specifiche:

1. *Generalizzazione*: la capacità di prevedere dati al di fuori del set di addestramento originale,
2. *Velocità di apprendimento*: l'aumento della velocità di convergenza, specialmente per i sistemi che apprendono da esperienze in tempo reale, e
3. *Tolleranza ai guasti*: la capacità di funzionare nonostante rumore o interruzioni [Medler, 1998, pp.89-90 trad. mia]⁸²⁰.

Un'efficace sintesi dei principi chiave della retropropagazione è stata offerta dallo stesso Werbos [Werbos, 1995, in Arbib, 1995], il quale richiama due aspetti centrali che stiamo prendendo in esame: da un lato la matematizzazione e la non serialità dell'apprendimento, resa possibile dall'adattamento dei valori dei pesi, per cui "la backpropagation è qualsiasi tecnica per adattare i pesi dei parametri di un sistema non lineare utilizzando in qualche modo tali derivate o l'equivalente"; dall'altro la natura neuromorfica delle reti neurali, giacché "la backpropagation è una procedura per calcolare in modo efficiente le derivate di una certa quantità in uscita di un sistema non lineare rispetto a tutti gli input e i parametri di quel sistema, attraverso calcoli che procedono all'indietro dalle uscite verso gli ingressi. Essa consente un'implementazione locale su hardware parallelo o su wetware" [Werbos, 1995, in Arbib, 1995, p.135, trad. mia]⁸²¹.

⁸²⁰ "Although basic backpropagation is powerful enough to solve a wide variety of problems, much work is done on improving the performance of artificial neural networks especially in regards to three specific characteristics:

1. *Generalization*: the ability to predict data outside the original training set,
2. *Learning Speed*: increasing the convergence rate, especially for systems learning from real-time experience, and
3. *Fault Tolerance*: the ability to perform despite noise or breakage".

⁸²¹ "Backpropagation is any technique for adapting the weights of parameters of a nonlinear system by somehow using such derivatives or the equivalent; Backpropagation is a procedure for efficiently calculating the derivatives of some output quantity of a nonlinear system with respect to all inputs and parameters of that system through

I risultati ottenuti con il metodo della retropropagazione dell'errore, insieme alle ricerche pionieristiche e isolate di studiosi come Kohonen e Grossberg, fornirono negli anni '80 – la cosiddetta “*Golden Age*” del connessionismo – una spinta decisiva alla rinascita degli studi sulle reti neurali, come paradigma alternativo e opposto all'architettura seriale dell'elaborazione dell'informazione. Questi sviluppi confluirono e contribuirono al successo di quella che è stata definita la “bibbia” del paradigma connessionista [Buckner, Garson, 2019, p. 79]: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* [McClelland, Rumelhart et al., 1986]. L'opera, in due volumi, curata da James L. McClelland (1948–) e David E. Rumelhart (1942–2011), con il contributo di Geoffrey E. Hinton (1947–) e di altri membri del cosiddetto “*PDP Research Group*”, attivo principalmente presso la Carnegie Mellon University e la University of California di San Diego, riuniva ricercatori provenienti dalla psicologia cognitiva, dalle scienze cognitive e dall'informatica.

Nei due volumi che compongono questo, “uno dei principali manifesti teorici del connessionismo” [Frixione, 1992, p.32] la “backpropagation teneva il centro della scena” [Buckner, Garson, 2019, p.79]⁸²². Accanto a questa, è la precisa ispirazione alla struttura fisiologica parallela dell'automa biologico⁸²³, ad aver sancito il successo del programma di ricerca che anima il testo.

Un motivo per cui i modelli PDP ci appaiono così attraenti, è l'evidenza del loro sapore «fisiologico»: *essi appaiono molto più legati alla fisiologia del cervello di tutti gli altri tipi di modelli di elaborazione delle informazioni*. Il cervello è costituito da un grande numero di elementi notevolmente interconnessi [...] che a quanto pare si inviano gli uni con gli altri dei messaggi eccitatori e inibitori semplicissimi, e aggiornano le loro eccitazioni in base a questi semplici messaggi [McClelland et. al, 1986, p.39 trad. it enfasi mia].

Nei modelli PDP come abbiamo visto analizzando la rete multi-layer⁸²⁴, in opposizione esplicita, riconosciuta dagli stessi autori, al paradigma seriale [Cfr. McClelland et. al, 1986,

calculations proceeding backwards from outputs to inputs It permits local implementation on parallel hardware or wetware”.

⁸²² “Backpropagation held center stage”.

⁸²³ Sulla plausibilità fisiologica della retropropagazione dell'errore si veda Arbib, 2003, p.144.

⁸²⁴ Un modello classico di addestramento di una rete PDP è quello della *Interactive Activation and Competition* (IAC), introdotto da James L. McClelland nel 1981 [McClelland, 1981] e successivamente confluito i PDP [McClelland et al., 1986, pp.57-63 trad. it.]. L'IAC mostrava come una rete di unità elementari, collegate da connessioni eccitatorie e inibitorie, potesse recuperare sia informazioni specifiche sia conoscenze generali a partire da rappresentazioni distribuite. L'esempio famoso è quello dei membri delle gang ipotetiche dei “*Jets*” e degli “*Sharks*”: ogni individuo è rappresentato da un nodo istanza connesso ai nodi che descrivono le sue proprietà

pp.39-42 trad. it.]⁸²⁵, “l’elaborazione delle informazioni ha luogo tramite le interazioni di un grande numero di elementi semplici di elaborazione, detti unità, ognuna delle quali manda segnali eccitatori od inibitori alle altre unità” [Cfr. McClelland et. al, 1986, p. 39 trad. it.].

Questo punto rimanda a un argomento di carattere cognitivo da affiancare a quello della somiglianza neuromorfica con il cervello e richiama le considerazioni centrali nello studio del sistema nervoso circa la natura dell’errore e del malfunzionamento (in integrazione con la teoria dell’informazione) nei confronti comparati tra calcolatori e automi biologici.

Nelle reti neurali, infatti, accanto all’argomento della maggiore fedeltà alla fisiologia cerebrale, emerge la dimensione cognitiva: esse mostrano una risposta ai danni e alla corruzione dell’informazione in ingresso molto più robusta e realistica rispetto ai computer classici. In questi ultimi, la perdita o la modifica casuale di unità provoca tipicamente un fallimento catastrofico (o nessun effetto, come nel caso della perdita di una locazione di memoria inutilizzata). Nei modelli connessionisti addestrati con *backpropagation*, invece, la rimozione di unità o la modifica casuale delle connessioni non blocca il funzionamento complessivo della rete, che continua a operare in maniera adeguata, seppure con minore accuratezza.

Questa *graceful degradation* delle funzioni – ossia la capacità di un sistema di continuare a operare in modo soddisfacente anche quando alcune connessioni risultano interrotte o errate, e di comportarsi in maniera sensata anche sulla base di dati parziali o contenenti errori – è considerata, ritornando all’intersezione tra livello cognitivo e quello neurofisiologico e computazionale, una caratteristica essenziale del cervello. Quest’ultimo riesce infatti a mantenere prestazioni complessive grazie alla ridondanza e alla distribuzione dei processi, continuando a operare anche in presenza di dati incompleti o di rumore [Cfr. Buckner, Garson, 2019, p. 79; Medler, 1998, p. 82; Clark, 1989, pp. 89-90].

Inoltre, i “sistemi dinamici complessi o non lineari, cioè sistemi che fanno emergere le loro proprietà dal comportamento collettivo di un grande numero di unità piuttosto semplici che hanno caratteristiche unicamente di tipo fisico” [Parisi, 1991, p.12], propri dei modelli

(età, stato civile, occupazione, ecc.), mentre le proprietà sono organizzate in gruppi mutuamente esclusivi e competono tra loro. Attivando un solo nodo (ad esempio il nome “Lance” o l’appartenenza alla gang “Jets”), la rete propaga l’attivazione e, attraverso l’equilibrio tra eccitazione e inibizione, è in grado di completare l’informazione mancante, recuperare caratteristiche tipiche e mostrare proprietà centrali delle reti connessioniste. Una spiegazione del modello accessibile si trova in Frixione, 1992, pp.33-38. Per l’approfondimento matematico si veda Medler 1998 79-82.

⁸²⁵ Per il confronto filosofico tra l’elaborazione seriale, che assume la mente come manipolatrice di simboli, e il “*Brain’s-eye view*”, che assume la struttura del cervello come modello della mente, Cfr. Clark, 1989, pp. 83-105.

connessionisti basati su reti neurali, non possono che essere trattati con un metodo statistico, a differenza della logica formale applicata a simboli nel paradigma cognitivista classico [Parisi, 1991, p.14], come abbiamo visto analizzando gli automi di von Neumann. Non può non ritornare all'attenzione il fatto di come Shannon avesse mostrato la natura stocastico-markoviana di un processo cognitivo “superiore” come il linguaggio⁸²⁶.

Come teoria dell'elaborazione dell'informazione che non riduce tutto alla mente vista come *software*, ma presta attenzione al design del *wetware* [McClelland et al., 1986, p. 31, trad. it.; Parisi, 1991, p. 10], il connessionismo – e il modello del *parallel processing* – presenta un set di otto proprietà fondamentali: - un insieme di unità di processamento, - uno stato di attivazione per ciascuna unità, - una funzione di output, - un pattern di connettività tra le unità, - una regola di propagazione che governa il modo in cui i pattern si trasmettono lungo la rete, - una regola di attivazione che combina gli input ricevuti con lo stato corrente producendo un nuovo livello di attivazione, - una regola di apprendimento che modifica i pattern di connettività con l'esperienza, e infine - un ambiente entro cui il sistema deve operare [Rumelhart et al., 1986, pp. 81-82].

⁸²⁶ Un processo markoviano descrive una sequenza in cui la probabilità del passo successivo dipende solo dallo stato presente e non dall'intera storia [Markov, 1906]. Shannon applicò questo schema all'inglese, trattandolo come una sorgente statistica [Shannon, 1945; 1948], e nelle Macy Conferences l'idea del linguaggio come processo stocastico fu ampiamente discussa. Al *Symposium on Information Theory* tenuto al MIT nel settembre 1956 – evento più volte ricordato come l'atto inaugurale delle scienze cognitive – Noam Chomsky presentò il saggio *Three Models for the Description of Language* [Chomsky, 1956]. In esso mostrò che né le grammatiche a stati finiti né le approssimazioni statistiche possono descrivere adeguatamente l'inglese, o una lingua naturale, poiché le dipendenze a distanza arbitraria e frasi grammaticali ma improbabili (*Colorless green ideas sleep furiously*) sfuggono a tali modelli. Chomsky propose quindi la *grammatica generativa*, fondata su strutture frasali, *kernel sentences* e trasformazioni ricorsive, ponendo le basi di un approccio formale al linguaggio naturale irriducibile a un mero processo probabilistico. [Chomsky, 1956, pp.113-124]. Riguardo al linguaggio, Rumelhart e McClelland [Rumelhart e McClelland, 1986] usarono la *backpropagation* per addestrare reti neurali a prevedere il passato dei verbi inglesi. Il compito era interessante perché, accanto alla regola regolare del suffisso *-ed*, molti verbi frequenti sono irregolari. La rete imparò bene il training set e si generalizzò anche a verbi nuovi, compresi alcuni irregolari. Durante l'apprendimento comparvero errori di sovraestensione (*break* → *broked*), simili a quelli dei bambini. Questo fu interpretato come realismo del modello, ma generò critiche [Pinker e Prince, 1988] sul fatto che le reti trattassero male alcuni verbi regolari nuovi, sollevando dubbi sulla capacità dei modelli connessionisti di catturare regole generali [Cfr. Rumelhart & McClelland, 1986 vol.2, pp. 216-268]. A riprova del ruolo pionieristico di Turing nel paradigma connessionista, in *Intelligent Machinery* [Turing, 1948] egli osserva che le macchine non organizzate di tipo-P, dotate di unità di memoria distribuite casualmente, potrebbero comportarsi in modo simile all'apprendimento umano dei verbi irregolari: addestrate a lungo, sarebbero capaci di applicare regole complesse senza saperle enunciare esplicitamente, proprio come avviene nell'uso del linguaggio. I modelli linguistici di grandi dimensioni (LLM) contemporanei possono essere visti come forme sofisticate di processi di Markov di ordine elevato, che operano su finestre di contesto precedenti e mancano di un modello interno esplicito della sintassi o della struttura gerarchica. In questo senso, sembrano segnare un ritorno alla concezione originaria di Shannon del linguaggio, sebbene presentino comportamenti patologici come ripetizione, incoerenza o degenerazione in determinate condizioni di campionamento [Zekri et al., 2024; Ildiz et al., 2024].

Queste caratteristiche corrispondono alle sei proprietà funzionali fondamentali del neurone stesso:

L'unità di processamento è il neurone stesso. Lo stato di attivazione e la regola di attivazione fanno parte del dispositivo di input e di integrazione del neurone, e la funzione di output non è altro che l'uscita del neurone. Il *pattern* di connettività e la regola di propagazione corrispondono invece alla funzione conduttiva del neurone. La regola di apprendimento e l'ambiente appartengono infine alle funzioni computazionali e rappresentazionali del neurone [Medler, 1998, pp.63 trad. mia]⁸²⁷.

L'approfondimento del funzionamento del sistema nervoso centrale nelle sue unità organizzative ha condotto all'elaborazione di un modello computazionale ispirato ad esso, capace di tradurre in pratica alcune intuizioni già presenti in von Neumann, legate alla statistica e alla teoria dell'informazione. È peculiare, in relazione alla vicenda del matematico ungherese, che le tecnologie del "paradigma PDP" abbiano avuto un paragone "esplosivo" riguardo alle loro potenzialità, a riprova della onnipresente natura sociotecnica e dello sviluppo "esternalista" delle tecnologie digitali e della computazione.

I risultati delle architetture connessioniste – sebbene lontanissimi da prestazioni di livello umano, o da quelle dell'IA contemporanea cui pure hanno contribuito a ispirare – indussero enti governativi e privati a riprendere a finanziare la ricerca sulle reti neurali. "Nel 1988, un alto funzionario della Defense Advanced Research Projects Agency (Darpa), il più munifico ente erogatore di finanziamenti all'IA, si esprime così: «Credo che questa tecnologia verso cui ci stiamo avventurando [ossia le reti neurali] sia più importante della bomba atomica»" [Mitchell, 2019, p. 29, trad. it.].

In quest'ottica, due risultati emblematici che possiamo collocare al livello di astrazione dell'implementazione sono la *Boltzmann Machine* e la *Connection Machine*.

La formulazione della prima macchina si deve a G. Hinton e Terry Joseph Sejnowski (1947–), neuroscienziato e informatico statunitense, [Hinton e Sejnowski, 1983]. Questa viene descritta come una classe di reti neurali stocastiche che adottano la tecnica della *temperatura simulata* (*simulated annealing*), mutuata dalla fisica statistica, per apprendere distribuzioni di probabilità complesse. Ogni unità è binaria (on/off; 0/1) e si aggiorna in modo probabilistico

⁸²⁷ The processing unit is the neuron itself The state of activation and the activation rule are part of the input and integrative device of the neuron and the output function is simply the output of the neuron The pattern of connectivity and propagation rule map onto the conductive function of the neuron And the learning rule and environment are part of the computational and representational functions of the neuron.

in base agli input ricevuti, cosicché lo stato globale della rete corrisponde a una configurazione caratterizzata da un certo livello di “energia”. L’evoluzione del sistema consiste nella ricerca di configurazioni via via più stabili, riducendo l’energia fino a raggiungere un equilibrio che segue la distribuzione di Boltzmann.

L’idea era di costruire un meccanismo non deterministico per l’attribuzione dei valori di verità, in cui le configurazioni più probabili (a energia più bassa) rappresentano stati più coerenti con i dati. Per evitare che il processo si blocchi in minimi locali, la rete è progettata per accettare talvolta anche transizioni verso stati a energia superiore, con una probabilità regolata da un parametro detto “temperatura”: man mano che la temperatura si abbassa, la dinamica del sistema converge verso configurazioni ottimali [Hinton, Sejnowski, 1983].

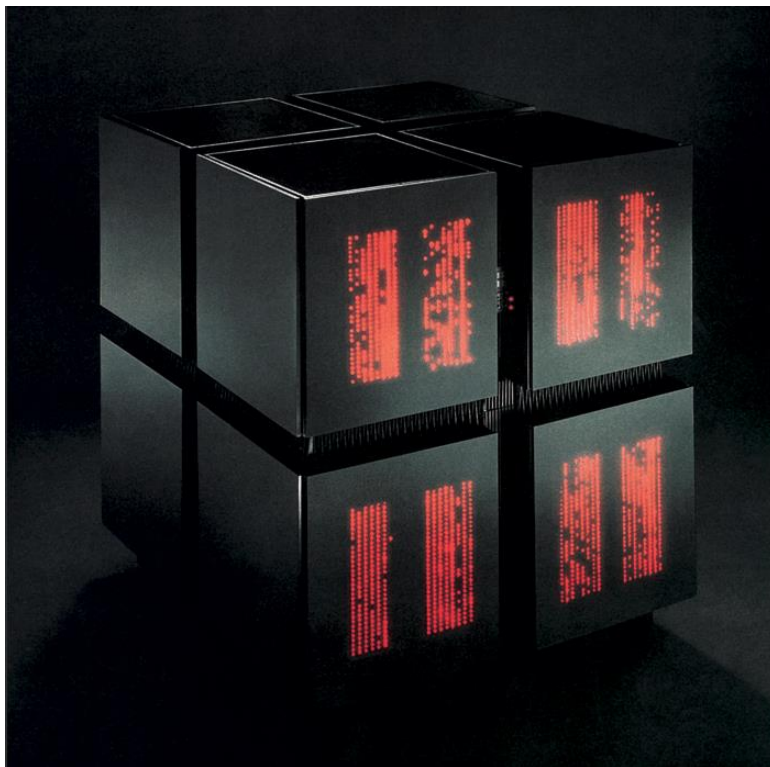
La macchina, pur essendo stata effettivamente implementata, ha incontrato notevoli difficoltà pratiche: il calcolo dei gradienti richiedeva un numero enorme di campioni, rendendo possibili solo reti di dimensioni ridotte. Questa limitazione ne ha ostacolato l’uso su larga scala, ma ha messo in risalto la natura stocastica, la dipendenza dalla distribuzione di Boltzmann e il ruolo della temperatura simulata come strumento per esplorare lo spazio delle architetture computazionali. Proprio da queste difficoltà nacque l’evoluzione verso modelli più semplici ed efficienti, come le *Restricted Boltzmann Machines* (RBM), reti neurali stocastiche semplificate che permisero l’addestramento gerarchico strato per strato delle architetture profonde [Hinton et al., 2006].

La seconda macchina in questione è la *Connection Machine*, una serie di supercomputer resa celebre anche dal film *Jurassic Park* di Steven Spielberg (1993) e dall’iconico design a “cubo di cubi” concepito dall’artista e designer statunitense Tamiko Thiel (1957–), ispirato alla topologia interna a ipercubo e animato da LED rossi che visualizzavano lo stato di funzionamento del sistema [Thiel, 1994][fig.29] .

Il progetto nacque dalla tesi di dottorato di D.W. Hillis (1957–) [Hillis, 1986], ingegnere e computer scientist statunitense, discussa nel 1985, e trovò poi uno sbocco imprenditoriale con la fondazione della Thinking Machines Corporation. La Connection Machine è stata “il primo calcolatore ad alto parallelismo ispirato ai principi della computazione neurale” (Frixione, 1992, p. 33). In queste sue caratteristiche si poneva come alternativa all’architettura classica di von Neumann, in un ulteriore superamento di quest’ultima maturato proprio all’interno del programma di ricerca che, prendendo a modello l’automa biologico del sistema nervoso, mirava a costruire nuove forme di elaborazione ispirate alla sua logica e organizzazione strutturale.

La Connection Machine nasce infatti come "un computer programmabile basato su automi cellulari", la cui idea fondamentale – e qui risuona l'assonanza con lo studio dei *cellular automata* – è che "una matrice cellulare regolare, con connessioni locali, possa essere fatta comportare come se le sue celle di elaborazione fossero collegate in qualsiasi topologia desiderata" [Hillis, 1984, trad. mia].

Fig.29 il display della Connection Machine



Hillis avvia la sua riflessione con un paragone tra la velocità e le prestazioni dei componenti del cervello e quelli del computer, osservando come, in modo paradossale, i transistor siano molto più veloci dei neuroni e, in teoria, i computer dovrebbero avere una potenza di calcolo 10.000 volte superiore al cervello. Eppure, nella pratica, accade l'opposto [Cfr. Hillis, 1986, pp. 2-3]. La causa non risiede nella velocità o nel numero dei componenti, bensì nell'uso inefficiente dell'architettura dei computer, che riflette "assunzioni obsolete" [Hillis, 1986, p.3 trad. mia]⁸²⁸. L'inefficienza del design seriale deriva dal "progetto bipartito",

⁸²⁸ ""Classical Computer Architecture Reflects Obsolete Assumptions".

nato dall'esigenza di mantenere i costosi tubi a vuoto costantemente occupati, e dunque dalla separazione "tra memoria da un lato ed elaborazione dall'altro" [Cfr. Hillis, 1986, p.4 trad. mia]⁸²⁹. "Questa inefficienza permane indipendentemente da quanto veloce si renda il processore, poiché la durata del calcolo diventa dominata dal tempo necessario a spostare i dati tra processore e memoria. Questo è il cosiddetto *collo di bottiglia di von Neumann*. Più grandi costruiamo le macchine, peggiore diventa" [Hillis, 1986, p.5 trad. mia enfasi mia]⁸³⁰.

La soluzione, allora, è nel rovesciare l'assunzione chiave del design seriale del calcolatore, e del modello che offre per la scienza cognitiva e la GOFAL – e realizzare una "una macchina di calcolo più omogenea, nella quale memoria ed elaborazione siano combinate" [Hillis, 1986, p.5 trad. mia]⁸³¹. La Connection Machine è un processore *general-purpose* a parallelismo massiccio, concepito per unire memoria ed elaborazione in un unico sistema: quando si assegna memoria a un dato, si assegna automaticamente anche la capacità di calcolarlo, così che l'elaborazione avvenga direttamente dove i dati risiedono [Cfr. Hillis, 1986].

Siamo così arrivati al cuore del paradigma *subsimbolico* - che raccoglie e traduce le riflessioni fatte finora a livello di studio degli automi e individuazioni di paradigmi alternativi di implementazione e di design computazionale (culminati nelle reti neurali) - e che si trova "tra il livello neuronale e quello concettuale" [Smolensky, 1988, p.78 trad. it]⁸³². Da questo punto di vista "le nostre scariche neurali si possono considerare subsimboliche, poiché si trovano alla base dei simboli che il nostro cervello in qualche modo crea" [Mitchell, 2019, p.20 trad. it.]

Secondo l'ipotesi del connessionismo subsimbolico, gli stati mentali sono definiti nei termini di relazioni funzionali fra oggetti computazionali astratti, e nei termini delle relazioni fra tali oggetti computazionali e il mondo. Ciò che caratterizza tale ipotesi è che non sempre esiste una corrispondenza precisa fra rappresentazioni mentali e questi oggetti computazionali: le unità subsimboliche devono essere considerate entità definite funzionalmente, che tuttavia (di per sé) non corrispondono ad alcuna

⁸²⁹ When von Neumann and his colleagues were designing the first computers, their processors were made of relatively fast and expensive switching components, such as vacuum tubes, whereas the memories were made of relatively slow and inexpensive components, such as delay lines or storage tubes. The result was a two-part design that kept the expensive vacuum tubes as busy as possible. We call this two-part design, with memory on one side and processing on the other, the von Neumann architecture, and it is the way that almost all computers are built today".

⁸³⁰ "His inefficiency remains no matter how fast we make the processor because the length of the computation becomes dominated by the time required to move data between processor and memory. This is called the von Neumann bottleneck. The bigger we build machines, the worse it gets".

⁸³¹ "The obvious answer is to get rid of the von Neumann architecture and build a more homogeneous computing machine in which memory and processing are combined".

⁸³² Una corrispondenza e il paragone tra le architetture neurali e quelle subsimboliche, cioè tra la "corteccia cerebrale" e i "sistemi dinamici connessionisti" si trova in Smolensky, 1988, p. 87 trad.it.

rappresentazione mentale. Le rappresentazioni mentali *emergono* come strutture di più alto livello distribuite su tali unità. [Frixione, 1992, p.46 enfasi mia]⁸³³.

4. *Going deep: Scenari tecnici e critici dell'IA contemporanea*

Il *Deep Learning* (DL) che anima la contemporaneità dell'Intelligenza Artificiale odierna si fonda su questa idea di rappresentazione subsimbolica: "Il deep learning risolve questo problema centrale dell'apprendimento delle rappresentazioni introducendo rappresentazioni espresse in termini di altre, più semplici rappresentazioni. Il deep learning consente al computer di costruire concetti complessi a partire da concetti più semplici" [Goodfellow, et. al, 2016, p.5 trad. mia]⁸³⁴.

Il fondamento di questa idea basilare dell'apprendimento profondo si ancora su quella di *rappresentazione distribuita* del connessionismo e dei modelli PDP: "Si tratta dell'idea secondo cui ciascun input a un sistema debba essere rappresentato da molte caratteristiche, e ciascuna caratteristica debba partecipare alla rappresentazione di molti possibili input" [Cfr. Goodfellow et al., 2016, p.18 trad. mia]⁸³⁵.

Le idee di rete neurale, di backpropagation, di superamento della distinzione gerarchica e seriale tra elaborazione e memoria e di rappresentazione distribuita - insieme all'integrazione della statistica e della teoria dell'informazione nella logica e nel design computazionale - viste attraverso la lente della metafora cibernetica a partire dalla riflessione di von Neumann sugli

⁸³³ Il *Deep Learning* (DL) contemporaneo si basa ancora su questa idea di *rappresentazione distribuita* del connessionismo e dei modelli PDP: "Si tratta dell'idea secondo cui ciascun input a un sistema debba essere rappresentato da molte caratteristiche, e ciascuna caratteristica debba partecipare alla rappresentazione di molti possibili input" [One of these concepts is that of distributed representation (Hinton et al., 1986). This is the idea that each input to a system should be represented by many features, and each feature should be involved in the representation of many possible inputs. Cfr. Goodfellow et al., 2016, p.18 trad. mia]. Sulla centralità, analizzata anche dal punto di vista tecnico, del concetto di rappresentazione nel DL

⁸³⁴ "Deep learning solves this central problem in representation learning by introducing representations that are expressed in terms of other, simpler representations. Deep learning allows the computer to build complex concepts out of simpler concepts".

⁸³⁵ "One of these concepts is that of distributed representation (Hinton et al., 1986). This is the idea that each input to a system should be represented by many features, and each feature should be involved in the representation of many possible inputs.". Sulla centralità, analizzata anche dal punto di vista tecnico, del concetto di rappresentazione nel DL, si rimanda a Goodfellow et al. 2016, pp.526-554. Una panoramica filosofica "globale" sul connessionismo e sulle reti neurali è in Di Francesco, 2003, pp.173-181. Altrettanto esaustiva è l'analisi dei problemi di filosofia della mente di questo paradigma delineata in Frixione, 1992, p.38-50, che analizza il problema alla luce delle prospettive *riduzioniste* o *eliminativiste* del mentale rispetto alla sfera della fisiologia del cervello. Classici del dibattito sul connessionismo (in relazione alle scienze cognitive simboliche). Una critica filosofica "classica" all'architettura connessionista è quella offerta in Fodor, Pylyshyn, 1988. Su parallelismo e reti neurali si vedano anche Bechtel, Abrahamsen, 1992 e 2002. L'idea di intelligenza senza livello di rappresentazione è affrontata, dal punto di vista della robotica, nell'affascinante Brooks, 1991

automi, sono tuttora alla base degli sviluppi dell'intelligenza artificiale. Prenderli in esame ci consentirà anche l'apertura di uno spiraglio analitico per indagare, data la natura sociotecnica di queste tecnologie, per alcune riflessioni di carattere critico ed etico.

Come riconosciuto nell'articolo pubblicato su *Nature* nel 2015 [LeCun et al., 2015] dai tre "padrini del deep learning" - Yann LeCun, Yoshua Bengio e Geoffrey Hinton - la backpropagation non è stata soltanto il principio che, negli anni Ottanta, ha permesso di superare i limiti dei perceptron e di rilanciare il connessionismo, ma è anche il motore concettuale e tecnico che ha reso possibile la rinascita del deep learning a partire dai primi anni Duemila. La possibilità di calcolare e propagare a ritroso gli errori in una rete multilivello ha consentito di addestrare architetture sempre più profonde, capaci di apprendere rappresentazioni interne complesse e di sostituire progressivamente le caratteristiche progettate a mano con quelle apprese automaticamente dai dati.

Su queste basi si svilupparono le *convolutional neural networks* (ConvNet), ispirate alla neurofisiologia visiva e fondate su connessioni locali, pesi condivisi e meccanismi di pooling. Le ConvNet sfruttarono la retropropagazione per addestrarsi in profondità, imparando gerarchie di caratteristiche: dai bordi agli oggetti, dai fonemi alle frasi, dai pixel alle scene [LeCun et al., 2015]

Dopo un periodo di relativa marginalità, la svolta giunse con la competizione ImageNet del 2012, quando la rete neurale convoluzionale AlexNet di A. Krizhevsky e I. Sutskever - dottorandi sotto la supervisione di G. Hinton - ridusse drasticamente l'errore rispetto ai metodi tradizionali. Quell'evento segnò l'inizio della rivoluzione del deep learning nella computer vision e, di lì a poco, in molte altre aree [Krizhevsky et al., 2012; Cfr. Mitchell, 2019, pp.73-90 trad. it.].

Un momento cruciale, quasi a chiudere simbolicamente un cerchio con il *Machine Learning* inaugurato dal programma per la dama di Samuel [Samuel, 1959], è stata la vittoria di AlphaGo, il programma sviluppato da Google DeepMind, nel marzo 2016.

AlphaGo riuscì a sconfiggere il campione mondiale di Go Lee Sedol in quello che, da sempre, era considerato il banco di prova più arduo per l'intelligenza artificiale, a causa dell'enorme vastità delle combinazioni possibili (oltre 10^{170} mosse legali). Il sistema combinava reti neurali profonde con la ricerca ad albero Monte Carlo ed era stato addestrato sia su un database di circa 30 milioni di mosse umane, sia attraverso migliaia di partite di auto-apprendimento, che gli consentivano di migliorare progressivamente le proprie prestazioni. AlphaGo non giocava come un maestro umano - non come il grande campione cinese Ke Jie,

allora numero uno al mondo – ma riuscì comunque a prevalere, segnando una svolta storica per l'IA [Silver et al., 2016; Russell, Norvig, 2021; Floridi, 2022, pp.51-52 trad. it].

Nel dicembre 2017 DeepMind presentò la sua evoluzione, AlphaZero, capace di imparare a giocare non più da partite umane ma unicamente dalle regole dei giochi e da milioni di partite generate contro sé stesso. Con questa forma di auto-apprendimento, basata su *deep reinforcement learning*, AlphaZero divenne in poche ore il giocatore più forte mai esistito: bastarono 9 ore per superare i migliori programmi di scacchi, 12 ore per lo shogi e 13 giorni per il Go. La novità cruciale stava nel fatto che il sistema si addestrava su *dati sintetici*⁸³⁶ prodotti in autonomia, sviluppando così strategie originali e non riconducibili ai motori tradizionali [Silver et al., 2018; Cfr Floridi, 2022, pp.70-72 trad. it.].

Un ulteriore traguardo di DeepMind è stato *AlphaFold*, presentato nel 2020, che ha affrontato uno dei problemi più complessi della biologia, quello del ripiegamento delle proteine, riuscendo a prevederne la struttura tridimensionale con una precisione senza precedenti [Senior et al., 2020].

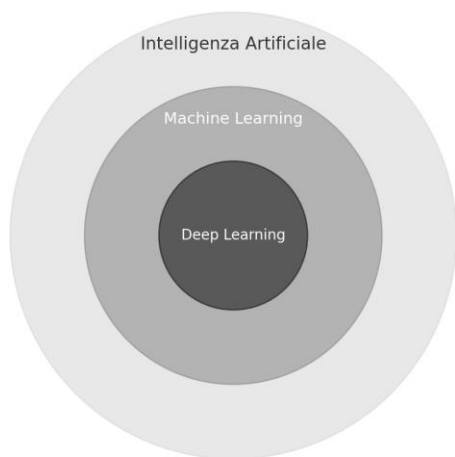
Nel 2017 si è imposto il principio per cui “*attention is all you need*”: con il *Transformer* [Vaswani et al., 2017] si è avviata una trasformazione nell'architettura delle reti neurali. Il cuore del modello è la *self-attention*, un meccanismo in cui a ogni elemento della sequenza vengono associati tre vettori: una *query* (che pone la domanda), una *key* (che funge da chiave di confronto) e un *value* (che contiene l'informazione). L'attenzione si calcola confrontando *query* e *key* per attribuire un peso di rilevanza e combinando i corrispondenti *value*. Con la variante a più teste (*multi-head attention*), il sistema coglie diversi tipi di relazioni in parallelo, riducendo la distanza computazionale tra posizioni anche lontane e migliorando le prestazioni delle reti neurali ricorrenti e convoluzionali. Da allora il Transformer si è imposto come modello di

⁸³⁶ Per *dati storici* si intendono quelli raccolti dal mondo reale (ad esempio cartelle cliniche, immagini, transazioni). Per *dati sintetici* si intendono invece quelli generati interamente da un sistema artificiale, senza alcun input diretto dal mondo reale (come nel caso di AlphaZero che apprende solo dalle regole e dalle proprie partite). I *dati ibridi* combinano elementi dei due, derivando in parte da dati storici e in parte da dati generati artificialmente. [Floridi, 2022, pp.69-77]. Un esempio di dati storici ibridizzati è quello fornito dai GAN (*Generative Adversarial Network*) in cui due reti competono: il *generatore* crea nuovi dati a partire da rumore casuale, mentre il *discriminatore* li confronta con i dati reali e segnala quanto siano convincenti; iterazione dopo iterazione, il generatore impara a produrre dati sintetici sempre più simili a quelli storici, usati soprattutto per immagini, video e altri contenuti visivi. Una spiegazione accessibile sui GAN si trova in Lee, Chen, 2021, pp.79-81. L'uso dell'AI – e dei GAN- nel mondo dell'arte e della produzione artistica, un ambito tradizionalmente *umano*, è trattato in a cura di Barale, 2020.

riferimento in campi diversi, dal *natural language processing* alla visione artificiale, fino alla generazione di contenuti.

Sino alla più recente contemporaneità, con il rischio costante di obsolescenza delle analisi, l'impatto dell'intelligenza artificiale - di cui il *deep learning* rappresenta soltanto un sottoinsieme [Fig. 30] - si è progressivamente ampliato fino a investire la quotidianità di ciascuno, dai motori di ricerca agli assistenti vocali, dai sistemi di traduzione automatica agli strumenti di gestione della crescente mole di informazione digitale. Architetture profonde con miliardi di parametri sono divenute la spina dorsale dell'IA contemporanea. Tale dinamica è emersa con particolare evidenza a partire dal lancio di *ChatGPT* da parte di OpenAI, nel novembre 2022, e con l'integrazione, persino nei motori di ricerca, di forme di *Intelligenza Artificiale Generativa*, ovvero “tecniche computazionali capaci di generare contenuti apparentemente nuovi e significativi, come testi, immagini o audio, a partire da dati di addestramento” [Feuerriegel et al., 2023, trad. mia]⁸³⁷.

Fig. 30 tipi di Learning e i sottoinsiemi dell'Intelligenza Artificiale



Il primo esempio di *machine learning* fu il programma per la dama di Samuel [Samuel, 1959], che utilizzava l'algoritmo *minimax* per scegliere le mosse migliori simulando in anticipo le possibili partite. L'apice dell'IA simbolica fu invece *Deep Blue*, che nel 1997 sconfisse Kasparov impiegando la cosiddetta

⁸³⁷ The term *generative AI* refers to computational techniques that are capable of generating seemingly new, meaningful content such as text, images, or audio from training data.

forza bruta, ossia la capacità di esplorare milioni di mosse al secondo grazie alla potenza di calcolo, combinata con funzioni di valutazione scacchistiche costruite a mano [Newborn, 2012]. Il deep learning, per contro, non si affida a regole predefinite, ma apprende dai dati secondo tre modalità principali: supervisionato, quando si allena su coppie input-output etichettate (ad esempio un'immagine accompagnata dall'etichetta "gatto" o "cane"); non supervisionato, quando scopre da solo strutture e raggruppamenti nei dati privi di etichette (ad esempio individuando somiglianze tra immagini senza sapere a priori cosa rappresentino); e per rinforzo, quando ottimizza le proprie azioni attraverso un sistema di ricompense e penalità, come accade nei giochi. In questo senso il deep learning rende possibile apprendere direttamente dalle esperienze e dai dati in modo adattivo [Russell, Norvig, 2021].

Se con Inverno dell'Intelligenza Artificiale [*AI Winter*] si intende il termine coniato nel 1984 durante l'American Association of Artificial Intelligence (oggi *Association for the Advancement of Artificial Intelligence*, AAAI), esso designa un ciclo di crisi, originato da promesse troppo ambiziose e aspettative puntualmente disattese dai risultati effettivi della ricerca; caratterizzato da un "effetto domino che prende avvio dal crescente scetticismo nella comunità scientifica, si propaga attraverso i mass media e infine coinvolge gli enti finanziatori" [Umbrello, 2021, trad. mia]⁸³⁸ oggi, al contrario, ci troviamo nel pieno di una vera e propria *estate torrida* dell'IA, come testimoniano i dati degli ultimi *Stanford AI Index Reports*.

Considerando soltanto gli investimenti privati globali in AI, si osserva un livello senza precedenti: nel 2021 si attestavano a 91,9 miliardi di dollari [Stanford AI Index Report 2022, p.70], per poi subire una leggera flessione nel 2022 a 86,1 miliardi [Stanford AI Index Report 2023, p.129] e nel 2023 a 85,1 miliardi [Stanford AI Index Report 2024, p. vii, Executive Summary]. Nel 2024, tuttavia, solo negli Stati Uniti si sono registrati investimenti pari a 109,1 miliardi, da cui si può stimare un valore globale compreso tra 135 e 140 miliardi [Stanford AI Index Report 2025, p. 30].

Ancora più indicativa della fase di espansione è la traiettoria degli investimenti in *Generative AI*. Dopo un livello relativamente contenuto di 1,5 miliardi di dollari nel 2021 [Stanford AI Index Report 2022, p.7]), il settore ha conosciuto un'impennata significativa: 3,6 miliardi nel 2022 [Stanford AI Index Report 2024, p.115], fino all'impressionante cifra di 28,6

⁸³⁸ "Coined in 1984 at the American Association of Artificial intelligence (now the Association for the Advancement of Artificial Intelligence or AAAI), the various boom and bust periods of AI research and funding lead AI researchers Marvin Minsky and Roger Schank to refer to the then-impending bust period as an AI Winter. Canadian AI researcher Daniel Crevier describes the phenomenon as a domino effect that begins with cynicism in the AI research community that then trickles to mass media and finally to funding bodies. The result is a freeze in serious AI research and development. This initial pessimism is mainly attributed to the overly ambitious promises that AI can yield with the actual results being far humbler than expectations".

miliardi nel 2023 - pari a una crescita del +694% - [Stanford AI Index Report 2025, p. 54], per poi raggiungere i 33,9 miliardi nel 2024, con un ulteriore incremento del +18,7% [Stanford AI Index Report 2025, p. 54].

Questi dati – e si noti il contrasto con i 13500 dollari richiesti da McCarthy e colleghi per la loro proposta di summer camp a Dartmouth nel 1956, dove venne coniato il termine *Artificial Intelligence*, o con i 2325 dollari concessi a Norbert Wiener per il lavoro sul predittore da cui scaturirà la cibernetica – mostrano con chiarezza come, a differenza delle fasi di gelo del passato, l'attuale congiuntura sia contraddistinta da una dinamica di crescita esplosiva, trainata in particolare dall'AI generativa e dal lancio pubblico di ChatGPT 3.5 nel novembre 2022, evento che ha segnato una svolta decisiva nella percezione e negli investimenti verso questa tecnologia.

Parallelamente, è cresciuto in maniera esponenziale – riproducendo quelle dinamiche già osservate nella diffusione storica del computer, tra paure distopiche di disumanizzazione e speranze di un'utopia delle macchine – il discorso pubblico e mediatico sull'intelligenza artificiale, a riprova del carattere irriducibilmente sociotecnico delle tecnologie che abbiamo analizzato.

Tale discorso si configura come sempre più pervasivo e polarizzato, oscillando tra aspettative messianiche e narrative catastrofistiche. Oggi, chiunque navighi sui social, segua un telegiornale o osservi l'uso (e l'abuso) di questi strumenti da parte di figure di primo piano della scena globale, sperimenta quotidianamente una comunicazione che, spesso attraverso contenuti generati dalle stesse tecnologie, si muove tra promesse prometeiche di un paradiso digitale – come quelle evocate dalle *Machines of Loving Grace* – e visioni apocalittiche ormai radicate nell'immaginario collettivo, alimentate anche da prodotti della cultura pop come *Black Mirror*, che mettono in scena futuri distopici dominati da algoritmi fuori controllo.

Un esempio emblematico di questa dialettica si ritrova nelle copertine e nei reportage delle principali testate internazionali. La rivista *TIME*, una delle più lette, influenti e iconiche al mondo, ha pubblicato il 12 giugno 2023 un numero speciale con una scelta grafica particolarmente evocativa: su un campo rosso acceso campeggiava, appena distorta da un glitch digitale, la scritta che dava titolo al reportage, *The End of Humanity: How Real Is the Risk?* Due mesi prima, il 13 aprile, Ian Hogarth – imprenditore, tecnologo e consigliere del governo britannico per l'intelligenza artificiale – aveva pubblicato sull'edizione online del *Financial Times* quello che è diventato l'articolo più letto della storia del sito, dal titolo inequivocabile: *We must slow down the race to God-like AI*.

A loro volta, il 29 marzo 2023, il Future of Life Institute – influente think tank statunitense dedicato a orientare lo sviluppo tecnologico secondo gli interessi a lungo termine dell'umanità e presieduto dal fisico svedese-americano Max Tegmark – ha diffuso una lettera aperta, datata 22 marzo, intitolata *Pause Giant AI Experiments: An Open Letter*. Firmata da oltre 33.000 persone, tra cui Elon Musk, il cofondatore di Apple Steve Wozniak, lo scienziato cognitivo e critico dell'IA Gary Marcus, Geoffrey Hinton (futuro premio Nobel per la fisica nel 2024, insignito per i suoi contributi fondamentali alle reti neurali artificiali) e il premio Turing Yoshua Bengio (riconosciuto, insieme a Hinton e Yann LeCun, come uno dei tre *godfathers* dell'intelligenza artificiale), la lettera proponeva una moratoria nello sviluppo dei modelli di intelligenza artificiale più avanzati. Il documento segnalava i rischi potenziali in termini occupazionali, di stabilità politica e di circolazione della disinformazione, invocando regolamentazioni più stringenti e un maggiore controllo democratico.

Il testo richiamava le preoccupazioni già avanzate alla *Conferenza di Asilomar sull'Intelligenza Artificiale Benefica* (California, 5–8 gennaio 2017), promossa dallo stesso istituto, dove lo sviluppo dell'intelligenza artificiale era stato definito un cambiamento profondo nella storia della vita sulla Terra, che dovrebbe essere previsto e gestito con cura e risorse proporzionate” [Future of Life Institute, 2023]⁸³⁹. Ma nello stesso documento fanno la loro comparsa affermazioni e domande di tutt'altro tono, come: “Dovremmo sviluppare menti non umane che potrebbero, col tempo, superarci in numero, in intelligenza, renderci obsoleti e sostituirci? Dovremmo rischiare di perdere il controllo della nostra civiltà?” [Future of Life Institute, 2023] evocanti la minaccia di un rischio esistenziale⁸⁴⁰ per la nostra specie rappresentato dalle nuove tecnologie

⁸³⁹ La lettera è disponibile su: <https://futureoflife.org/open-letter/pause-giant-ai-experiments/>. Sullo stesso sito del Future of Life Institute il documento completo della conferenza di Asilomar dei 23 principi enunciat si trova sullo stesso sito all'indirizzo <https://futureoflife.org/ai-principles>.

⁸⁴⁰ Un rischio è detto esistenziale (*existential risk*, o *X-risk*) quando “minaccia di provocare l'estinzione delle forme di vita intelligenti originarie della Terra o di compromettere in modo permanente e drastico il loro potenziale futuro di sviluppo” [Bostrom, 2014, trad. it.]. Secondo Nick Bostrom, filosofo di Oxford e advisor del Future of Life Institute, tra questi rischi – per cui ha proposto una tassonomia [Bostrom, 2002] – rientra la possibile comparsa di una Superintelligenza artificiale, il cui sviluppo, se non orientato da intenti benevoli verso l'umanità, potrebbe risultare catastrofico. Anche un sistema progettato per uno scopo banale, come fabbricare graffette (nel celebre esperimento mentale del *paperclip maximizer*), porterebbe avanti il suo compito sino a convertire tutta la materia del cosmo, compresa quella organica di cui è fatto l'essere umano, in graffette [Bostrom, 2003]. Riflessioni di questo genere sono al centro di una filosofia oggi particolarmente *à la page* nella Silicon Valley: il *Longtermism* (Lungotermismo) [Signorelli, 2023]. Questo prende le mosse dall'*Effective Altruism* (Altruismo effettivo), una prospettiva etica sviluppatasi negli anni '10 del XXI secolo, ispirata anche alle teorie di Peter Singer sul superamento del particolarismo morale [Singer, 1972]. Essa promuove l'uso dell'evidenza scientifica, di analisi costi-benefici e del ragionamento per fare il massimo bene possibile, concentrando le risorse su cause ritenute ad altissimo impatto [MacAskill, 2017]. Il Longtermism radicalizza la prospettiva dell'*Effective Altruism*, estendendo gli obblighi morali – nella sua versione forte – nel tempo, verso le generazioni di umani di un futuro a lunghissimo

Accanto alle evocazioni di scenari catastrofistici, si staglia – come contraltare – una visione utopica che attribuisce agli sviluppi più recenti dell'intelligenza artificiale e delle tecnologie digitali la possibilità di rappresentare una sorta di panacea per i mali della società contemporanea. Questa narrazione salvifica è spesso promossa dagli stessi protagonisti della Silicon Valley e talvolta rilanciata, quasi senza soluzione di continuità, dalle medesime testate che nei giorni precedenti paventano scenari di rovina. Ancora *TIME* ne offre un esempio significativo: l'11 novembre 2024 pubblicava sul proprio sito un articolo dal titolo eloquente, *A Roadmap to AI Utopia*, in cui si immagina un futuro in cui l'IA genera “abbondanza senza precedenti”, rivoluziona sanità ed educazione e promette l'avvento di una società post-scarità⁸⁴¹.

L'aver seguito l'*informazione*⁸⁴² – e con essa la metafora cibernetica dell'analogia funzionale tra automa biologico e artificiale, con la trasformazione in senso stocastico della loro logica computazionale – mette in luce un ulteriore aspetto. L'architettura di von Neumann, come abbiamo visto, oltre a derivare dallo stato dell'arte della tecnologia disponibile, rispondeva a istanze di natura socio-tecnica: il calcolatore e la retorica che lo sorreggeva si

termine. Si possono leggere nei testi chiave del Longtermism affermazioni del tenore per cui salvare una vita in un paese povero potrebbe avere effetti indiretti inferiori rispetto a farlo in un paese ricco, dove produttività e innovazione sono più alte [Ord, 2020]. Agire nel presente sul pianeta e sui suoi abitanti – la lotta al cambiamento climatico, alle disuguaglianze o alle malattie endemiche – sarebbe un obiettivo etico meno prioritario rispetto allo sviluppo del “vast and glorious” potenziale della razza umana, sia esso la colonizzazione di galassie lontane, in cui potrebbero vivere miliardi di esseri umani, sia lo sviluppo – in una prospettiva di potenziamento transumanistico – attraverso le tecnologie digitali. Testi chiave, oltre a quelli già citati della prospettiva lungotermista, sono MacAskill, 2022; Tegmark, 2018. È facile comprendere perché il Lungotermismo sia stato definito dal filosofo Emil P. Torres, nella sua efficace critica al Longtermism (Torres, 2021), che ne denuncia anche le dinamiche di interesse economico, “quite possibly the most dangerous secular belief system in the world today”. Si veda anche Signorelli, 2023. Esponenti lungotermisti come Ord o James Matheny sono stati consiglieri nei governi e nelle amministrazioni statunitensi e britanniche e in organizzazioni sovranazionali come l'OMS o la Banca Mondiale. Si capisce anche perché imprenditori e esponenti della tecno-destra della Silicon Valley siano stati promotori e finanziatori degli istituti e dei centri di ricerca lungotermisti. Un ulteriore esempio di radicalizzazione è la nuova, e se possibile maggiormente pericolosa, filosofia dell'Accelerazionismo Efficace (*Effective Accelerationism*, abbreviato *e/acc*), fortemente influenzata dalle idee del filosofo Nick Land. Secondo questa corrente, che ribalta e al contempo radicalizza la logica alla base del Longtermism, la spinta tecnologica dovrebbe essere accelerata senza vincoli. Lo sviluppo tecnologico estremo, possibile causa di X-risk, ma i fautori dell'*e/acc* – o auspicano e mirano semmai ad accelerarlo, per conquistare il prima possibile la società post-umana che considerano – costi quel che costi – il prossimo necessario passaggio evolutivo. Un passaggio che mira anche a trasformare il capitalismo delle democrazie liberali in un anarcocapitalismo privo di ogni regolamentazione, scrupolo egualitario e tutela nei confronti dei più deboli”. Tutto ciò va di pari passo con il rifiuto di qualunque forma di regolamentazione o di controllo etico, considerati inutili ostacoli che rischiano di rallentare lo slancio dell'umanità verso il prossimo stadio evolutivo” [Signorelli, 2025]. Sulla permeabilità delle élite tecnologiche californiane a prospettive tecno-escatologiche hanno probabilmente influito anche le, già incontrate nel nostro percorso dinamiche contro-culturali, che hanno intravisto nello spazio digitale e nelle nuove tecnologie un nuovo territorio, libero da regole e costrizioni.

⁸⁴¹ L'articolo, scelto tra gli svariati altri dello stesso tenore che è facilissimo trovare in rete, è disponibile all'indirizzo <https://time.com/7174892/a-roadmap-to-ai-utopia/>

⁸⁴² Una storia esaustiva del concetto di “informazione” si trova in Capurro, Hjørland, 2003.

inscrivevano pienamente nel “mondo chiuso” [Edwards, 1996], nato dopo la seconda guerra mondiale e consolidatosi nella guerra fredda.

A questo quadro si possono affiancare alcune considerazioni che evidenziano la differenza tra il paradigma “classico” dell’intelligenza artificiale e delle scienze cognitive e quello sviluppatosi nella ricerca sul connessionismo e sulle reti neurali - entrambi di comune origine cibernetica e, al pari delle considerazioni sociotecniche già svolte a proposito di quest’ultima, qui ulteriormente rafforzate dall’esame delle modalità di implementazione software/hardware e di analisi computazionale. La cibernetica, infatti, e la retorica che l’ha accompagnata hanno innescato sin dall’inizio una tensione tra, da un lato, spinte alla liberazione e alla personalizzazione e, dall’altro, tendenze a ridurre l’uomo “al livello di esecutori degli ordini di un centro nervoso che pretende di essere superiore” [Wiener, 1950, p. 30, trad. it..]

Riguardo a questo rapporto tra paradigmi computazionali e condizioni socioculturali, abbiamo visto come l’intelligenza artificiale cognitiva “classica” si fondi sul primato della dimensione logico-razionale linguistica (come manipolazione di simboli), una prospettiva non neutrale e con implicazioni pratiche. Scriveva a riguardo il filosofo e scienziato cognitivo italiano D. Parisi, scomparso nel 2021:

Questa fondamentale impostazione pratica si inserisce in un lungo processo storico di progressiva razionalizzazione delle attività umane che risale agli inizi della tradizione culturale dell’Occidente e ha tappe importanti nella filosofia utilitaristica anglosassone del ’700 e dell’ ’800, nell’emergere della scienza economica, con la sua idea dell’«homo oeconomicus» come perseguitore razionale di scopi e di interessi, nell’accelerazione dei processi di organizzazione, razionalizzazione e burocratizzazione della vita economica e sociale dalla fine del secolo scorso ad oggi. L’informatica e l’intelligenza artificiale sono una estensione lineare di questi sviluppi: i programmi di azioni umane pensati a tavolino, valutati nei loro effetti, controllati nella loro esecuzione, diventano programmi di istruzioni, e quindi poi di azioni, di una macchina. Da questo punto di vista la figura di Herbert A. Simon, premio Nobel per l’economia, studioso di scienza dell’amministrazione, e uno dei padri dell’intelligenza artificiale, è emblematica [Parisi, 1988, pp.124-125].

Per quanto riguarda invece l’orizzonte sub-simbolico e connessionista, anche in questo caso sembra emergere, secondo le significative riflessioni di Parisi, un legame tra la crisi dell’IA

simbolica classica, l'affermarsi di paradigmi alternativi come quelli connessionisti e il corrispondente mutamento del contesto socioeconomico, politico e culturale⁸⁴³:

Vi è poi la ragione che ho già ricordato, una ragione di clima culturale e socio-politico cambiato. Non so quanto peso si possa dare a queste cose, ma ho l'impressione che per certi aspetti vi sia maggiore omogeneità oggi tra alcuni orientamenti di fondo della società, della politica e della cultura e i modelli connessionisti che non tra questi orientamenti e i modelli dell'intelligenza artificiale. La planetarizzazione crescente del pianeta, cioè la sua progressiva interconnettività globale, appare più facilmente interpretabile con modelli non centralizzati e non razionali, ma piuttosto basati su continui aggiustamenti dinamici e su soluzioni globali frutto di molte interazioni locali, come sono i modelli connessionisti. Lo stesso vale per la dinamica socio-politica e anche più generalmente culturale all'interno dei singoli paesi, con il ruolo decrescente del governo e dello stato, con l'incertezza e la pluralità dei valori e degli stili di vita, e così via" [Parisi, 1988, pp.267-268].

Alle cautele espresse dalle riflessioni di Parisi, possiamo aggiungere quelle già richiamate, legate all'eredità del significato culturale del seme cibernetico sbocciato nelle tecnologie computazionali: la riduzione dell'essere umano a insiemi di dati e pattern predittivi, le problematicità e i *bias* della generalizzazione statistica che appiattisce le differenze individuali, la visione a black box che oscura i processi interni e la conseguente opacità dei processi decisionali automatizzati.

⁸⁴³ Possiamo notare una differenza anche nella dimensione pratica, legata al modello imprenditoriale e all'organizzazione aziendale, tra l'IA simbolica e quella connessionista. Come mostra Thomas Haigh, negli anni Sessanta il calcolatore venne presentato come destinato a realizzare il *management information system* (MIS), concepito come il futuro dell'informatica aziendale e del management stesso. Si trattava di un progetto ambizioso, ma in gran parte irrealizzabile, che prendeva forma dalle aspirazioni già coltivate negli anni Cinquanta dai cosiddetti *systems men*: presentare il computer non più come mero strumento di *data processing*, ma come *information system*, attraverso il quale essi cercavano di legittimarsi come mediatori privilegiati del controllo manageriale. In questo quadro, l'uomo era ridotto principalmente al ruolo di fornitore o esecutore di dati, all'interno di una concezione fortemente centralizzata e gerarchica. Nonostante il fallimento tecnico del progetto, l'influenza delle idee ispiratrici del MIS ha continuato a esercitare un ruolo duraturo, arrivando fino agli anni Novanta con i concetti di *enterprise resource planning* (ERP) e *business intelligence*, e costituendo tuttora una delle modalità dominanti di concepire il rapporto tra tecnologie digitali e organizzazione [Haigh, 2001]. Diversamente, Francisco J. Varela (1946–2001), biologo e filosofo cileno noto anche per i suoi studi con Humberto R. Maturana (1928–2021) sull'autopoiesi dei sistemi viventi e cognitivi, nelle sue ricerche successive sulla scienza della cognizione ha mostrato come il paradigma connessionista proponga un modello decentrato e auto-organizzativo, in cui l'ordine emerge dalle interazioni locali tra le componenti e non da un centro di controllo. Trasposto sul piano dell'organizzazione, ciò significa che la gestione non consiste più nella raccolta e pianificazione centralizzata delle informazioni, ma nella creazione delle condizioni perché si sviluppino connessioni distribuite, pattern collettivi e processi emergenti. Il cambio di paradigma, dunque, non ha solo trasformato la concezione della cognizione, ma anche l'immaginario organizzativo, aprendo la via a modelli più flessibili, adattivi e meno gerarchici [Varela, 1987].

L'aver individuato nella prospettiva della metafora cibernetica una lente analitica per ripercorrere la genealogia di questi processi ci ha offerto una prospettiva privilegiata per la presa in esame dello sviluppo delle tecnologie della computazione e digitali che, dopo aver, per così dire, "esternalizzato" alcune capacità eccedenti quelle umane nel calcolatore, e dopo il tentativo di riprodurre nelle macchine le caratteristiche della struttura cognitiva umana (con l'architettura di von Neumann, lo *stored program principle* e lo sviluppo della programmazione), ha subito un'ulteriore evoluzione, legata all'universo di dati analizzato in precedenza. Come sostiene T. Numerico:

La prospettiva cibernetica si è evoluta negli ultimi anni grazie alla diffusione della rete Internet che ha consentito l'archiviazione nei data center delle grandi piattaforme di tutte le attività che avvengono nel contesto digitale. L'intelligenza artificiale si è trasformata, così, nella costruzione di algoritmi in grado di prevedere, anticipare e orientare il comportamento umano, dopo averlo discretizzato in un flusso di dati misurabili. Non si tratta più di una simulazione dell'agire umano, ma di una sua riproduzione, dopo averlo reificato in forma di tracce leggibili. La replica digitale serve per anticipare abitudini e preferenze delle persone, e per valutare e orientare le loro attività future. La riproduzione che gli algoritmi di *machine learning* e *deep learning* propongono non riguarda l'essere umano sia come oggetto di analisi sia in quanto decisore e interprete delle informazioni disponibili [Numerico, 2021, p.16].

Lo sviluppo di un nuovo paradigma, maturato dall'approfondimento della metafora biologica, riporta la riflessione a uno dei temi chiave della cibernetica wieneriana: il progressivo *blurring* tra entità biologiche e artificiali, considerate dal punto di vista informazionale e analizzabili secondo modelli stocastici. In questa prospettiva, la gestione e l'elaborazione dell'informazione, così come il suo accesso pressoché ubiquo, risultano sempre più immediati, offrono un ausilio costante ai processi cognitivi e diventano parte integrante dell'attività di elaborazione umana, inscritta in un'architettura di *information processing* ibrida e diffusa, ma richiedono al tempo stesso un esame critico dei meccanismi computazionali che la supportano.

Se i "computer di von Neumann" – nelle direttrici storiche della loro implementazione – restano legati a un immaginario di utopia/distopia digitale, lo stesso matematico ungherese mise in evidenza l'irriducibile storicità del linguaggio logico e matematico. Gli algoritmi vanno considerati come vere e proprie "macchine culturali": "complessi aggregati di astrazioni, processi e persone" che affondano "le loro radici non solo nella logica matematica ma anche

nelle tradizioni filosofiche della cibernetica, della coscienza e della magia del linguaggio simbolico” [Finn, 2017, p. viii, trad. it.]. Le modalità della loro implementazione e analisi devono sempre tener conto di questa dimensione critica: “Comprendere come conoscerla richiede i metodi critici delle discipline umanistiche” [Finn, 2017, p. viii, trad. it.].

Abbiamo già visto, a partire dal lavoro di Turing, come la nozione stessa di algoritmo implichi un’interazione tra la procedura rigorosa di calcolo e le facoltà umane, e come lo stesso matematico abbia mostrato i limiti della sola “disciplina” computazionale con le sue macchine non organizzate. La tesi di Church-Turing, a sua volta, secondo Emil Post (1897–1954) – che giunse indipendentemente a risultati equivalenti [Post, 1936] – va assunta come una “legge naturale” riguardante non tanto una capacità, quanto piuttosto un limite: “una fondamentale scoperta circa le limitazioni del potere matematizzante dell’Homo sapiens” [Post, 1936, n. 8, trad. it. in Frixione, 1992, p. 15].

Da questa prospettiva, e avendo considerato non solo la dimensione storica e culturale degli algoritmi ma anche la loro struttura matematica, computazionale e implementativa, possiamo oggi mettere a fuoco alcune criticità delle forme assunte dalle tecnologie connessioniste contemporanee.

Queste, come denuncia il neuroscienziato e psicologo cognitivo Gary Marcus (1970–) nel suo articolo *Deep Learning: A Critical Appraisal* [Marcus, 2018], presentano problematicità che richiedono estrema attenzione critica riguardo alla loro diffusione e al loro utilizzo. Tra queste, in primo luogo, la dipendenza massiccia dai dati, dovuta alla loro logica statistico-induttiva. Il Deep Learning è infatti “*affamato di dati*” [*data hungry*], e la sua efficacia è proporzionale alle dimensioni del dataset disponibile: “Nei problemi in cui i dati sono limitati, il deep learning spesso non rappresenta una soluzione ideale” [Marcus, 2018, p. 6, trad. mia]⁸⁴⁴.

Oltre a ciò, emergono difficoltà sostanziali nell’*inferenza aperta* [*open-ended inference*], ovvero nella capacità di trarre conclusioni che vadano oltre le informazioni esplicite fornite dai dati: i sistemi riescono bene in compiti di domanda-risposta chiusi, ma falliscono quando è richiesta un’integrazione di conoscenze implicite o di background. A questo limite si collega la *manca di integrazione con la conoscenza pregressa*: i modelli tendono a imparare in modo isolato, senza incorporare saperi già consolidati. Così, in compiti che coinvolgono la fisica elementare o l’ingegneria, non partono dalle leggi di Newton, ma ricostruiscono statisticamente

⁸⁴⁴ “In problems where data are limited, deep learning often is not an ideal solution”

le regolarità osservando grandi quantità di dati grezzi, giungendo soltanto ad approssimare regole che per l'essere umano risultano già acquisite.

Un ulteriore limite è rappresentato dall'incapacità di distinguere causalità da correlazione: "il deep learning apprende complesse correlazioni tra input e output, ma senza una rappresentazione intrinseca della causalità" [Marcus, 2018, p. 12, trad. mia]⁸⁴⁵ Infine, persiste il problema dell'opacità: i modelli di deep learning sono modelli di *black box*, rendendo difficile comprendere come vengano prese le decisioni, con conseguenze potenzialmente gravi in ambiti come medicina, credito o giustizia. Per queste ragioni Marcus conclude "finora il deep learning funziona bene come approssimazione, ma le sue risposte non possono essere pienamente affidabili" [Marcus, 2018, p. 13, trad. mia]⁸⁴⁶.

Da questo punto di vista, gli algoritmi di *Deep Learning* alla base dei Large Language Models possono essere considerati dei "pappagalli stocastici" (*stochastic parrots*) – ossia sistemi che generano testo plausibile senza comprenderne realmente il contenuto, combinando parole e frasi sulla base di probabilità statistiche – come li hanno definiti Bender, Gebru, McMillan-Major e Mitchell nel loro articolo, ancora oggi al centro del dibattito, *On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?* [Bender et. al, 2021].

Un primo aspetto denunciato dalle autrici riguarda i costi ambientali e finanziari [*enviromnemntal and financial cost*]. L'addestramento di LLM richiede una quantità enorme di energia e risorse computazionali, con impatti diretti sul cambiamento climatico. Come sottolineano, "'Gli effetti negativi del cambiamento climatico stanno raggiungendo e colpendo per prime le comunità più marginalizzate del mondo.'" [Bender et al., 2021, p. 612 trad. mia]⁸⁴⁷. Inoltre, i costi proibitivi concentrano lo sviluppo di queste tecnologie nelle mani di poche grandi aziende e istituzioni, rafforzando squilibri economici e di potere [Cfr. Bender et al., p.613]⁸⁴⁸

⁸⁴⁵ "Roughly speaking deep learning learns complex correlations between input and output features, but with no inherent representation of causality". Non deve quindi sorprendere che, commentando la tendenza degli algoritmi di *Machine Learning* e di *Deep Learning* a individuare correlazioni tra enormi quantità di dati – che non corrispondono a relazioni causali, ma "rispondono piuttosto all'instaurazione di un legame di causa-effetto" [Numerico, 2021, p. 122] – T. Numerico richiami esplicitamente il nome di David Hume, attualizzando la riflessione del filosofo scozzese sull'induzione e sui problemi di correlazione e causalità come criterio epistemico di cautela rispetto alle inferenze predittive degli algoritmi [Numerico, 2021, p. 168].

⁸⁴⁶ "Deep learning thus far works well as an approximation, but its answers often cannot be fully trusted".

⁸⁴⁷ "The negative effects of climate change are reaching and impacting the world's most marginalized communities first"

⁸⁴⁸ In stretta relazione con questo tema vi è quello della *digital sovereignty*, intesa come capacità di esercitare un controllo effettivo e democratico sul digitale - dati, software, protocolli, infrastrutture, servizi - non solo sulla loro presenza ma anche sulla loro creazione, funzionamento e direzione. Si tratta di uno scontro asimmetrico tra grandi aziende, che detengono il potere creativo e operativo, e stati, cui resta soprattutto la funzione regolatoria. Negli

In secondo luogo, viene evidenziato il problema dei *dataset incommensurabili e scarsamente documentati*. [*unfathomable training data*] [Beder et al., p.614] La costruzione di corpora testuali di dimensioni gigantesche, provenienti in gran parte dal web, comporta ciò che le autrici definiscono un vero e proprio “*documentation debt*” ossia l'impossibilità di rendere trasparenti le fonti e di rendere tracciabili i contenuti. Questo deficit mina la possibilità di accountability e di controllo critico dei modelli. [Cfr Bender et. al, p.615],

Da qui discende una terza criticità, quella dei *rischi sociali e degli usi malevoli*. Gli LLM incorporano e amplificano bias e stereotipi preesistenti: “I modelli linguistici che producono testo riprodurranno e anzi amplificheranno i pregiudizi presenti nei loro dati di input.” [Bender et al, 2021, p. 617]. Le conseguenze non sono solo culturali, ma anche psicologiche – “I danni psicologici derivanti da microaggressioni e dalla minaccia degli stereotipi” [Bender et al. 2021, p. 617]⁸⁴⁹ – e possono tradursi in vere e proprie discriminazioni. A ciò si aggiunge la possibilità di impieghi dannosi, LLM come GPT3 – antecedente al modello di chatgpt lanciato nel novembre 2022 - usato per generare testi cospirazionisti a supporto del reclutamento estremista [Cfr. Bender et al. 2021, p. 617].

Infine, le autrici insistono sul tema dell'illusione di comprensione. Il testo generato da un pappagallo stocastico “non è fondato su un intento comunicativo, né su un modello del mondo, né su un modello dello stato mentale del lettore. Non potrebbe esserlo, perché i dati di addestramento non hanno mai incluso la condivisione di pensieri con un interlocutore, né la macchina ha la capacità di farlo” [Bender, 2021, p.616 trad. mia]⁸⁵⁰. In altri termini, la fluidità sintattica e la plausibilità semantica inducono a proiettare significato dove in realtà non vi è comprensione⁸⁵¹: un rischio che si traduce in casi concreti di *automation bias* e fraintendimenti. Emblematico è l'episodio di un uomo palestinese arrestato dalla polizia israeliana dopo che un suo post su Facebook, che in arabo recitava semplicemente “*good morning*”, era stato tradotto

ultimi decenni è emersa così una forma di sovranità de facto delle multinazionali, fondata su monopoli e autoregolazione, che ha trasformato gli individui da “*voter-consumer*” a “*follower-user*”. Per l'Europa la questione è distinguere tra una sovranità digitale nazionale, con il rischio di scivolare in sovranismo o statismo, e una sovranità digitale sovranazionale, più adeguata a bilanciare il potere delle piattaforme globali. Resta però aperto il problema della legittimazione democratica: come estendere la sovranità popolare, tradizionalmente ancorata allo Stato, al livello europeo. In questo senso, la sovranità moderna-analogica appare insufficiente e occorre affiancarle una sovranità digitale, capace di completarla e rafforzarla [Floridi, 2020].

⁸⁴⁹ “The psychological harms of microaggressions and stereotype threat”

⁸⁵⁰ “Text generated by an LM is not grounded in communicative intent, any model of the world, or any model of the reader’s state of mind. It can’t have been, because the training data never included sharing thoughts with a listener, nor does the machine have the ability to do that”

⁸⁵¹ A questo proposito, L. Floridi ha parlato di “*pareidolia semantica*”, ossia la tendenza a proiettare intenzionalità e comprensione su sistemi di IA che operano unicamente tramite correlazioni statistiche e pattern matching, inducendoci a vedere significato dove in realtà non vi è comprensione, Floridi, 2025.

automaticamente come “*hurt them*” in inglese e “*attack them*” in ebraico [Cfr. Bender et al., 2021, p. 618].

Alle criticità sopra elencate, e ad esse collegata, va aggiunta la tendenza, endemica [Zu et al., 2024]⁸⁵², e che ciascuno di noi sperimenta nell’uso dell’IA, degli LLM e degli algoritmi di deep learning, a sviluppare *allucinazioni* (*hallucinations*). Con questo termine si intende la generazione di contenuti che appaiono plausibili nella forma ma che non corrispondono alla realtà, oppure che non possono essere verificati rispetto al contesto o alle basi di conoscenza disponibili. In altre parole, il modello produce risposte inesatte, inventate o fuorvianti, che possono contraddire i fatti [*factuality hallucination*] o deviare dalle istruzioni e dal contesto fornito [*faithfulness hallucination*] [Huang et al., 2024]⁸⁵³.

Queste considerazioni critiche e tecniche contribuiscono a scardinare la narrazione utopistica e le speranze prometeiche delle nuove tecnologie. Altrettanto è possibile fare per la prospettiva catastrofista, quella per cui le macchine, diventate autonome e superintelligenti⁸⁵⁴, “potrebbero schiavizzare l’umanità come specie inferiore, ignorarne i diritti e perseguire i propri fini, indipendentemente dagli effetti che ciò ha sulle vite umane” [Floridi, 2022, p.260 trad. it.]. Daniel Dennett esemplificava queste questioni, un tempo relegate ad una dimensione fantascientifica ma che con la pervasività delle tecnologie digitali acquista via via concretezza, con la domanda “*When HAL kills, who's to blame?*” [Dennett, 1997].

Dalla nostra analisi sono emersi via via, e secondo diversi punti di vista, le dinamiche dell’interazione dell’essere umano con le tecnologie dell’informazione, e i limiti dei diversi paradigmi dell’IA che aspirano “sia a *riprodurre i risultati* o *l’esito positivo* del nostro comportamento intelligente (o almeno qualche tipo di comportamento animale) con mezzi non biologici, sia a produrre l’equivalente non biologico della nostra intelligenza, cioè la *fonte* di tale comportamento” [Floridi, 2022, p.48 trad. it.]. Possiamo, con Floridi, chiamare il primo approccio “ingegneristico” e il secondo “cognitivo” [Floridi, 2022, p.48 trad. it. enfasi originale].

Come brevemente mostrato dall’analisi della configurazione attuale dell’IA, “come settore della *scienza cognitiva interessata alla produzione dell’intelligenza*, l’IA rimane

⁸⁵² Già nel 1995 Stephen Thaler – ingegnere e ricercatore statunitense nel campo delle reti neurali e della *computational creativity* – aveva messo in luce come il fenomeno delle allucinazioni fosse intrinseco ai modelli connessionisti. In un semplice esperimento su una rete di associazione di pattern, dimostrò che, attraverso il *pruning* (ossia la rimozione casuale di connessioni sinaptiche artificiali), la rete tendeva a produrre output addestrati anche in assenza dei corrispondenti input. I pesi eliminati lasciavano infatti “vuoti” interpretati dagli strati nascosti e di output come se fossero segnali reali, generando così dei cosiddetti *virtual inputs*. [Thaler, 1995].

⁸⁵³ Si rimanda al medesimo per un survey complessivo del concetto di *hallucination*.

⁸⁵⁴ Su questo tema, e su quello della “singolarità”, si rimanda a Shanahan, 2015 e Floridi, 2022, pp.259-277 trad. it..

fantascienza e una triste delusione. L'IA *produttiva* non si limita a prestazioni inferiori rispetto all'intelligenza umana: non ha ancora preso parte alla competizione" [Floridi, 2022, p.49 trad. it. enfasi originale]. Al contrario, e ne è prova la straordinaria diffusione di queste tecnologie che avvolgono sempre di più il nostro ambiente di vita quotidiana, come "settore dell'*ingegneria*, interessata alla *riproduzione del comportamento intelligente*, l'IA ha avuto un successo sbalorditivo, ben oltre le più rosee aspettative" [Floridi, 2022, p.48 trad. it. enfasi originale].

Sin dai vari aspetti che abbiamo analizzato del lavoro di Turing, appare chiaro che intelligenza umana e comportamento intelligente, del quale con la diffusione delle tecnologie computazionali e digitali non abbiamo più il monopolio, vanno *scissi*. Come concepire allora le tecnologie ICT e l'IA?

La classica definizione controfattuale e l'interpretazione dell'IA come divorzio, e non come matrimonio, tra l'agire e l'intelligenza, consentono di concepire l'IA come una risorsa crescente di capacità di agire interattiva, autonoma e spesso autoapprendente (nel senso dell'apprendimento automatico [...] che può affrontare un numero sempre più elevato di problemi e attività che richiederebbero altrimenti l'intelligenza e l'intervento umani (e possibilmente una quantità illimitata di tempo) per essere eseguiti con successo. In breve, l'IA è definita sulla base di risultati e azioni ingegnerizzati e quindi, nel resto di questo libro, tratterò l'IA come una *riserva di capacità di agire a portata di mano* [Floridi, 2022, p.53 trad. it. enfasi mia].

A conferma di questa concezione dell'IA - che mette fuori gioco la retorica "apocalittica" sulle macchine - si può richiamare la prospettiva, dove si mostra nuovamente la proficuità dell'approccio "cibernetico" che abbiamo adottato, che mira a ricomprendere gli enti artificiali, e la loro *agency*⁸⁵⁵, nell'alveo della considerazione morale.

Ad un determinato Livello di Astrazione, infatti, un agente può essere descritto attraverso tre proprietà fondamentali: l'*interattività*, ossia la capacità di agire e subire l'azione del proprio ambiente, come avviene ad esempio nell'immissione o nell'emissione di un valore o in un'azione simultanea agente-paziente paragonabile alla forza gravitazionale tra corpi; l'*autonomia*, cioè la possibilità di cambiare stato senza rispondere direttamente a uno stimolo esterno ma attraverso transizioni interne, condizione che implica l'esistenza di almeno due

⁸⁵⁵ Su questo tema, e sul tema della crescente autonomia delle macchine e la loro trattazione etica si veda Tamburrini, 2020.

stati e che conferisce all'agente un certo grado di indipendenza dall'ambiente; e infine l'*adattabilità*, vale a dire la capacità di modificare, in seguito alle interazioni, le regole di transizione con cui cambia stato, così da poter essere considerato, a un dato Livello di Astrazione, come un ente che apprende dal proprio modo di operare in funzione della propria esperienza. Se le regole di transizione sono memorizzate come parte dello stato interno dell'agente, l'*adattabilità* può essere vista come derivante dalle altre due proprietà [Floridi, Sanders, 2004; Floridi, 2009c, p. 112].

Un agente – biologico, sociale, artificiale, ibrido – è tale se soddisfa queste condizioni minimali. “Un’azione può essere definita come moralmente qualificabile, se e soltanto se, può generare bene o male morale. Un agente può essere definito agente morale se è capace di azione moralmente qualificabile” [Floridi, 2009c, p.118]. Questa definizione di agente e di azione morale si svincola da una visione classica dell'imputabilità morale, “non richiede che l'agente possenga stati mentali come credenze o intenzioni ovvero capacità come il “libero arbitrio” (qualsiasi cosa si possa intendere per tale termine controverso)” [Bynum, 2009, p.20]. Un agente artificiale, da questa prospettiva, può essere imputabile anche se, in quanto privo di “mente”, non responsabile [Bynum, 2009, p.20].

La scissione tra *mens rea* e *actus reus* nella definizione di agente e azione morale [Cfr. Floridi, 2022, pp. 184-186 trad. it.], parallela a quella tra intelligenza e comportamento intelligente, è un ulteriore segno del “*potere di scissione* del digitale” capace di “scindere e ricomporre realtà e idee che abbiamo ereditato dalla modernità” [Floridi, 2022, p.21 trad. it. enfasi originale]. Tra queste idee che vengono “re-ontologizzate” e “re-epistemologizzate” [Floridi, 2022, p.31 trad. it.] vi sono quelle legate alla soggettività moderna, alla *res cogitans* cartesiana o all'io trascendentale kantiano, considerata come unica detentrica di quelle caratteristiche di autonomia, intenzionalità, libertà che lo qualificavano come agente moralmente responsabile⁸⁵⁶.

⁸⁵⁶ Da un punto di vista differente, quello della storia della scienza e dell'antropologia storica, il carattere “innaturale” di una certa concezione della soggettività è stato messo in luce da de Ceglia [2023, p. 106]. Sul superamento di tale concezione possono essere richiamate le riflessioni di Bruno Latour in *Non siamo mai stati moderni* [Latour, 1991]. Qui Latour mette in evidenza il legame tra la dimensione culturale e politica della soggettività, eredità della modernità, e la dimensione computazionale, chiamando in causa uno dei “nonni” dell'IA, Hobbes. Scrive infatti: “[Hobbes] a tutte le sue conclusioni scientifiche non arriva con l'opinione, l'osservazione o la rivelazione, ma con una dimostrazione matematica, l'unico metodo di argomentazione capace di costringere chiunque a dare il suo assenso; e a questa dimostrazione non arriva con calcoli trascendentali, come il re di Platone, ma con uno strumento di calcolo puro, il cervello meccanico, un computer ante litteram. Perfino il famoso contratto sociale non è che il risultato di un calcolo cui arrivano tutti insieme i cittadini terrorizzati che cercano di affrancarsi dallo stato di natura” [Latour, 1991, p. 33, trad. it.]. L'era in cui il digitale trasforma la soggettività moderna, aprendo la strada a entità artificiali e ibride, è forse quella in cui, come suggeriva Latour, “è tempo, forse, di parlare di nuovo di democrazia, ma di una democrazia estesa alle cose stesse” [Latour, 1991, p. 146, trad. it.].

Da questa punto di vista, un ente - biologico, artificiale o ibrido - può essere considerato un agente se soddisfa il LoA minimale dell'essere interattivo, adattabile o autonomo. In altri termini, se è in grado di partecipare al *moral game*⁸⁵⁷ sulla base delle proprie azioni, e non in virtù della presenza, o *assenza*, di una mente, esso va riconosciuto come agente morale.

Il quadro sin qui tracciato è perfettamente compatibile con la prospettiva informazionale - "cibernetica" - assunta come criterio analitico in tutto il corso di questo lavoro. In uno scenario come quello contemporaneo, in cui la diffusione di agenti - le cui caratteristiche minimali per definirli tali richiamano da vicino le condizioni di "ultrastabilità" dell'omeostato di Ashby - si presenta in modo sempre più pervasivo a livello personale, lavorativo e sociale, questo:

Rende più agevole la discussione della moralità degli agenti non soltanto nel cyberspazio ma anche nell'infosfera - in cui gli animali possono essere considerati agenti morali senza dover esibire una libera volontà, emozioni o stati mentali³⁶ - e in ciò che abbiamo definito come contesti di 'moralità distribuita', in cui agenti sociali e legali possono qualificarsi ora come agenti morali. Il grande vantaggio consiste nella migliore comprensione del discorso morale in contesti non umani. Il solo 'costo' di un approccio alla 'moralità non basato sul mentale' è l'estensione della classe degli agenti e degli agenti morali, per includere gli AA. È un costo che vale sempre di più la pena di pagare quanto più rapidamente ci muoviamo verso una società dell'informazione avanzata [Floridi, 2009c, p.131].

Un antecedente diretto di questa posizione è lo stesso Wiener. Pur partendo dall'assunto che tutti i sistemi sono regolati da meccanismi di feedback di processo dinamico dell'informazione, egli ipotizzò che le macchine del futuro sarebbero state in grado non solo di prendere decisioni e portarle a compimento, ma persino di apprendere dall'esperienza e di modificare i propri comportamenti sulla base di essa. Proprio per questo, mise in guardia dal

⁸⁵⁷ In Allen et al. 2000 si studia la possibilità di un test, ispirato all'*Imitation Game* di Turing, volto a valutare la capacità morale delle macchine. Il test proposto, denominato *Moral Turing Test* (MTT), consiste nel verificare se un agente artificiale sia in grado di comportarsi in modo moralmente indistinguibile da un essere umano, per un giudice umano, in situazioni eticamente significative [Allen et al., 2000]. La questione che emerge è se sia possibile considerare accettabile che le macchine debbano essere in grado di mentire riguardo a questioni di rilevanza etica, concernenti cioè la loro capacità di discernere e produrre bene e male, e quindi di evitare o arrecare danni ad altri enti, incluso l'essere umano. Questa idea, secondo cui nella possibilità di mentire si troverebbe la base per lo sviluppo di una moralità di tipo umano da parte degli agenti artificiali, è presente anche nel summenzionato saggio di Dennett su HAL. Posto che le macchine, per sviluppare una propria eticità, debbano accedere a una intenzionalità di ordine superiore [*high-order intentionality*], il prezzo da pagare per tale intenzionalità superiore sarebbe la duplice capacità di ingannare: tanto gli altri quanto se stessi. ["Another price we pay for higher-order intentionality is the opportunity for duplicity, which comes in two flavors: self-deception and other-deception", Dennett, 1997].

rischio che tali capacità potessero tradursi in conseguenze eticamente problematiche: macchine autonome, concepite con leggerezza o secondo gli errori tipici dell'“apprendista stregone”, avrebbero potuto sfuggire al controllo umano e agire secondo valori non condivisi, sostituendo il giudizio umano con decisioni potenzialmente pericolose [Cfr. Bynum, 2009, p.10]. Scriveva Wiener già nel 1950:

Qualsiasi macchina costruita per indicare delle decisioni, se non possiede la facoltà di imparare, agirà sempre in conformità di uno schema meccanico. Guai a noi se la lasceremo decidere della nostra condotta senza aver prima studiato le leggi che governano il suo comportamento, e senza sapere con certezza che questo comportamento sarà basato su principi che noi possiamo accettare! D'altra parte, come il genietto della bottiglia, la macchina che può imparare e può prendere decisioni sulla base di tale conoscenza acquisita, non sarà in alcun modo obbligata a decidere nello stesso senso in cui avremmo deciso noi stessi, o per lo meno in modo a noi accettabile. Per colui che non avrà coscienza di ciò, addossare il problema della propria responsabilità alla macchina (sia che questa possa apprendere oppure no) vorrà dire affidare la propria responsabilità al vento e vedersela tornare indietro fra i turbini della tempesta. [Wiener, 1950, p.228 trad. it.]

Ecco perché è importante adottare una prospettiva che sia capace di integrare anche dal punto di vista etico⁸⁵⁸ i sempre nuovi sviluppi tecnologici. “Come suggeriva preoccupato Wiener, rischiamo di essere lasciati fuori dal feedback perché troppo lenti” [Numerico, 2021, p.16]. Il padre della cibernetica comprese il ruolo via via crescente delle macchine nella società:

La società può essere compresa soltanto attraverso lo studio dei messaggi e dei mezzi di comunicazione relativi ad essi; e che, nello sviluppo futuro di questi messaggi e mezzi di comunicazione, i messaggi fra l'uomo e le macchine, fra le macchine e l'uomo, e fra macchine e macchine sono destinati ad avere una parte sempre più importante” [Wiener, 1950, pp.23-24 trad. it.]

Solo comprendendo questo, e coltivando “l'esistenza di un'intima relazione tra l'attività di processare informazioni inerente all'essere umano e lo *scopo* della vita umana” [Bynum, 2009, p.10], le tecnologie digitali possono essere uno strumento per la promozione di un *uso umano degli esseri umani*.

5. ***Design for Action: verso un'etica progettuale***

⁸⁵⁸ “Wiener comprese che, per prevenire tale genere di disastro, il mondo avrebbe avuto bisogno di regole etiche per gli agenti artificiali” [Bynum, 2009, p.11]

Le riflessioni etiche qui avanzate, radicate nell'analisi storico-tecnologica e storico-scientifica che ha attraversato l'intero lavoro, e sviluppate attraverso la lente informazionale e la "metafora cibernetica" con i suoi successivi sviluppi, consentono di individuare nel *design* delle tecnologie digitali lo spazio privilegiato per delineare sviluppi auspicabili, lontani tanto dai sensazionalismi utopici quanto da quelli distopici.

Il quadro contemporaneo, segnato dalla pervasività delle ICT e dalla diffusione crescente di sistemi di machine learning e deep learning - che radicalizzano quel processo avviato con le simulazioni del metodo Monte Carlo e con la progressiva costruzione di una "realtà computerizzata" - impone di spostare l'attenzione sulla fase della progettazione, nella quale si inscrivono possibilità d'uso, forme di interazione e vincoli di comportamento e un mutato quadro etico che coinvolge tutti i soggetti (umani e non umani) della relazione con le tecnologie, visti, dalla prospettiva che abbiamo adottato come entità informazionali.

Come abbiamo messo in mostra, le tecnologie digitali non sono meri sviluppi frutto di un determinismo tecnologico, ma sono artefatti sociotecnici, "assemblaggi di cose, persone, pratiche e significati" [Johnson, Wetmore 2021, p.97, trad. mia]⁸⁵⁹. Decisivo in questo intreccio è ciò che "riguarda i valori umani e sociali e il modo in cui essi sono incorporati nelle scelte tecnologiche. I valori plasmano la tecnologia e, a loro volta, ne sono plasmati" [Johnson, Wetmore, 2021, p.4 trad. mia]⁸⁶⁰. Gli artefatti, mediando l'azione, *materializzano* i valori [Verbeek, 2006]⁸⁶¹.

I *designer*, mediante l'uso di *affordances* e di vincoli (di natura fisica, logica, semantica e culturale) possono facilitare, sconsigliare, proibire, ostacolare o suggerire determinati usi di ogni tipo di oggetto, dalle maniglie delle porte di casa alle centrali nucleari [Norman 1988]. È ormai celebre il caso delle sopraelevate di Long Island, progettate da Robert Moses (1888–1981), urbanista e "master builder" di New York, che tra gli anni Venti e i Sessanta diresse la

⁸⁵⁹ "In order to avoid thinking of technology merely as physical objects, it is useful to think of technology as sociotechnical systems-assemblages of things, people, practices, and meanings".

⁸⁶⁰ "One of the most important aspects of this relationship centers around human and social values and how they are embedded in technological choices. Values shape and are shaped by technology".

⁸⁶¹ Un contributo decisivo all'analisi del concetto di mediazione tecnologica è stato quello del filosofo olandese Peter-Paul Verbeek, che, muovendosi nella cornice della postfenomenologia e a partire dal concetto di *script* di Latour, ha mostrato come la mediazione sia costitutiva dell'azione: gli artefatti non sono strumenti neutrali, ma plasmano modalità d'uso e forme di vita. Il design iscrive prescrizioni che non solo influenzano le azioni individuali, ma organizzano anche relazioni sociali e ambientali. Per questo Verbeek attribuisce una responsabilità condivisa a progettisti e utilizzatori, articolata nella necessità di ridurre gli impatti negativi, promuovere usi responsabili e sviluppare consapevolezza etica sulle conseguenze della mediazione tecnologica. In tale prospettiva, il design "realizza materialmente la moralità", poiché le tecnologie non si limitano a mediare azioni già date, ma co-determinano le forme dell'agire e, quindi, la responsabilità morale dei designer [Verbeek, 2006].

costruzione di strade, parchi e infrastrutture. Moses fece costruire i cavalcavia delle sue parkways a un'altezza volutamente ridotta, così da impedire il transito degli autobus, usati dalle fasce povere e dalle minoranze etniche della città, verso le spiagge destinate alla middle class bianca. Una scelta apparentemente tecnica, quella dell'altezza dei ponti, si tradusse così in un meccanismo di esclusione sociale e razziale, incorporato nella struttura stessa dell'infrastruttura [Winner, 1980].

All'inizio degli anni 2000, lo sviluppo del digitale ha spinto a studiare le modalità in cui i computer, il Web o qualsiasi altro oggetto o prodotto computazionale o informatico interattivo possono agire come tecnologie *persuasive*, capaci, cioè, di far compiere o non compiere agli utenti determinate azioni certe azioni, modificandone comportamento e atteggiamenti. "Definisco tecnologia della persuasione (o tecnologia persuasiva) qualsiasi sistema informatico interattivo progettato per modificare gli atteggiamenti o i comportamenti delle persone" [Fogg, 2003, p.16 trad. it]⁸⁶².

Da questa prospettiva, l'estrema pervasività delle tecnologie digitali odierne, che rende la nostra esperienza di vita sempre più *onlife*, ossia che "trascende i limiti della distinzione tra ambienti online e offline" [Floridi, 2014, p.83 trad. it.]. La possibilità, data la potenza computazionale degli artefatti, la frequenza e l'intensità dell'interazione con essi, di cadere in *dark patterns* - e cioè quelle pratiche progettuali "in cui i designer utilizzano la loro conoscenza del comportamento umano (ad es. psicologia) e dei desideri degli utenti finali per implementare funzionalità ingannevoli che non sono nell'interesse dell'utente." [Gray et al., 2018, trad. mia]⁸⁶³ - rende di strettissima urgenza valutare come e quali valori incorporare negli algoritmi di Machine e Deep Learning dell'IA.

⁸⁶² Il termine *captology* - derivato dall'acronimo *Computers As Persuasive Technologies* - è usato per indicare lo studio dei computer come tecnologie persuasive, ossia l'insieme delle discipline che esplorano in che modo strumenti interattivi (software, Web, dispositivi digitali) possono essere progettati per influenzare gli atteggiamenti e i comportamenti degli utenti. Nel 2003 B. J. Fogg pubblicò il volume *Persuasive Technology: Using Computers to Change What We Think and Do*, che pose le basi teoriche del campo [Fogg, 2003].

⁸⁶³ "We use the term *dark patterns* to define instances where designers use their knowledge of human behavior (e.g., psychology) and the desires of end users to implement deceptive functionality that is not in the user's best interest". Il termine *dark pattern* è stato introdotto dal designer britannico Harry Brignull nel 2010, per descrivere interfacce utente deliberatamente progettate per indurre gli utenti a compiere azioni contrarie ai propri interessi, sfruttando conoscenze psicologiche e bias cognitivi. Successivamente, Gray et al. 2018 hanno proposto una sistematizzazione individuando cinque principali categorie: *nagging*, ossia l'interruzione continua che devia l'utente dal compito desiderato; *obstruction*, il rendere un'azione (come disiscriversi) difficile o onerosa; *sneaking*, l'occultamento di informazioni rilevanti, come costi aggiuntivi; *interface interference*, la manipolazione grafica o visiva dell'interfaccia per privilegiare certe scelte; e *forced action*, l'imposizione di azioni obbligatorie, ad esempio la condivisione di dati personali come prerequisito per accedere a un servizio [Gray et al., 2018].

In questo senso, una sintesi unitaria dei principi elaborati nelle principali dichiarazioni internazionali sui principi dell'IA è stata proposta da Luciano Floridi, che individua cinque criteri fondamentali per orientare lo sviluppo di un'IA etica [2022, pp. 91-105]. Il principio di *beneficenza* richiama l'esigenza di promuovere il benessere e di perseguire il miglioramento della condizione umana attraverso l'impiego delle tecnologie. A questo si affianca il principio di *non maleficenza*, che esige di prevenire o almeno ridurre al minimo i danni che potrebbero derivare dall'uso degli artefatti digitali. Il principio di *autonomia* sollecita il rispetto delle capacità decisionali e della libertà di scelta degli individui, evitando forme di manipolazione o di coercizione. Il principio di *giustizia* riguarda la distribuzione equa dei benefici e dei rischi legati all'impiego dell'IA, e invita a vigilare affinché non vengano riprodotte o accentuate discriminazioni e disuguaglianze sociali. A questi quattro principi, quelli "classici" della biotica, Floridi ne aggiunge un quinto, specificamente "abilitante" per l'IA: il principio di *esplicabilità*, che "include sia il senso epistemologico di intelligibilità, sia il senso etico di responsabilità" [Floridi, 2022, p. 101, trad. it.].

Materializzando questi valori negli algoritmi, è possibile puntare allo sviluppo di una Intelligenza Artificiale per il Bene Sociale (*Artificial Intelligence for Social Good*, AI4SG) definita come "il design, lo sviluppo e l'implementazione di sistemi di IA in modo da (i) prevenire, mitigare o risolvere i problemi che incidono negativamente sulla vita umana e/o sul benessere del mondo naturale e/o (ii) consentire sviluppi preferibili dal punto di vista sociale e/o sostenibili dal punto di vista ambientale" [Floridi, 2022, p. 227]. In questa prospettiva, Floridi individua dei fattori essenziali che si traducono in altrettante *best practices* per guidare i designer la progettazione etica degli algoritmi⁸⁶⁴.

⁸⁶⁴ Le sette *best practices* individuate da Floridi per implementare l'AI4SG, in relazione ai 5 principi, sono: la *falsificabilità con implementazione incrementale*, che invita i progettisti a individuare requisiti verificabili e a testarli in fasi progressive, dall'ambiente di sviluppo al mondo esterno, così da assicurare affidabilità e correggere errori (*beneficenza, non maleficenza*); le *garanzie contro la manipolazione dei predittori*, che richiedono di prevenire l'uso di indicatori non causali e di limitare, quando appropriato, la conoscenza di come gli input influenzano gli output, così da evitare manipolazioni (*beneficenza, non maleficenza*); l'*intervento contestualizzato in favore del destinatario*, che prescrive di modellare l'intervento sulle caratteristiche, preferenze e finalità degli utenti, rispettando la loro autonomia e la possibilità di rifiutare o modificare le proposte (*beneficenza, autonomia*); la *spiegazione contestualizzata e finalità trasparenti*, che richiede ai designer di scegliere un livello di astrazione adeguato, fornire argomenti persuasivi e assicurare che lo scopo del sistema sia conoscibile per impostazione predefinita da chi ne subisce gli effetti (*beneficenza, spiegabilità*); la *tutela della privacy e il consenso dell'interessato*, che stabilisce che il trattamento dei dati debba avvenire nel rispetto della dignità, delle soglie contestuali di consenso e dei diritti fondamentali degli individui (*beneficenza, autonomia, non maleficenza*); l'*equità concreta*, che impone di ridurre le distorsioni dovute a variabili irrilevanti e garantire giustizia negli esiti, salvo quando l'uso mirato di certi dati sia necessario a favorire inclusione o sicurezza (*beneficenza, giustizia*); infine la *semantizzazione adatta all'umano*, che sollecita la progettazione di sistemi capaci di preservare e rafforzare la

Particolare attenzione, proprio dal punto di vista informativo che ha guidato la nostra analisi è la tutela del bene etico della *privacy*. Se in passato essa era intesa principalmente come diritto all'anonimato o a essere lasciati in pace, oggi si configura piuttosto come garanzia dell'integrità personale dei soggetti, intesi come entità informazionali che costruiscono la propria identità nelle relazioni con ambienti digitali. In questa prospettiva, la *privacy* non è più soltanto protezione da intrusioni esterne, ma condizione di possibilità della presenza stessa degli individui nello spazio dell'infosfera e di esercizio di controllo sulla propria identità, nelle nuove modalità di essere "presenti" che le tecnologie digitali rendono possibili, al di là della mera presenza fisica. Proteggere la *privacy* dei soggetti significa mantenerne l'integrità ontologica come pattern di dati e informazioni. [Cfr. Floridi 2009d, pp.132-154; Floridi, 2014, pp.115-146 trad.it; Floridi; 2019, pp.64-88 trad. it.; Floridi, 2025]⁸⁶⁵.

A riprova della rilevanza non solo tecnologica, ma anche sociale e politica che investe i temi del design e delle nuove tecnologie, in particolare in relazione alla *privacy*, occorre sottolineare come il legislatore, sia nazionale sia sovranazionale, abbia posto al centro dell'attenzione la dimensione etica nella progettazione delle nuove tecnologie. In questo orizzonte, la normativa ha progressivamente assunto un ruolo decisivo, culminato nell'adozione del *Regolamento Generale sulla Protezione dei Dati* (GDPR) dell'Unione Europea, entrato in vigore nel 2018, che ha introdotto principi fondamentali per la gestione e la tutela dei dati personali. Tra le innovazioni principali, poi recepite nelle legislazioni nazionali, si distinguono i principi di *Privacy by Design* e *Privacy by Default*, definiti rispettivamente all'art. 25 e nel considerando 78 del Regolamento.

Il principio di *Privacy by Design* stabilisce che la protezione dei dati personali debba essere incorporata direttamente nel design delle tecnologie: ciò significa che la tutela della *privacy* deve essere integrata sin dalle prime fasi di ideazione e sviluppo di sistemi, processi, prodotti o servizi, diventando parte costitutiva della loro architettura tecnica e concettuale. A esso si affianca il principio di *Privacy by Default*, che prescrive che i sistemi siano configurati in modo da garantire automaticamente il massimo livello di protezione possibile, senza richiedere interventi da parte dell'utente. Le impostazioni predefinite devono pertanto minimizzare la raccolta e l'uso dei dati, rafforzando la tutela dei diritti fondamentali delle persone.

capacità delle persone di attribuire senso alle informazioni e alle interazioni con l'IA, favorendo un dialogo fra esseri umani e macchine (*beneficenza, autonomia*) [Cfr. Floridi, 2022, pp. 231-258].

⁸⁶⁵ Per una panoramica sull'implementazione della *privacy* tecnologica, che sfata i miti sulla protezione dei dati attraverso algoritmi, si veda Hoepman, 2021. Per una trattazione snella ed efficace delle strategie di progettazione orientate alla *privacy*, si rimanda a Hoepman, 2022.

Quanto sancito dal GDPR riflette una crescente consapevolezza normativa circa la necessità di affrontare le sfide etiche sollevate dalle tecnologie digitali e dagli algoritmi, assicurando che la privacy, e più in generale i valori etici, siano incorporati fin dalla fase di progettazione delle tecnologie stesse.

Il quadro tracciato finora mostra come l'attenzione nei confronti delle nuove tecnologie si sia progressivamente spostata dalla dimensione dell'intelligenza a quella dell'azione. Questo passaggio ha comportato una ridefinizione della fenomenologia morale, segnata anche dalla proliferazione degli *stakeholder* coinvolti nei processi tecnologici: le distinzioni tra agenti umani, artificiali o ibridi, pazienti morali, tecnologie che mediano l'agire e progettisti degli artefatti diventano sempre più porose, e la sfera etica coinvolge tutti i poli di questa relazione, inscrivendo la responsabilità direttamente nei processi di design.

Il quadro tracciato finora mostra come l'attenzione nei confronti delle nuove tecnologie si sia progressivamente spostata dalla dimensione dell'intelligenza a quella dell'azione. Questo passaggio ha comportato una ridefinizione della fenomenologia morale, in cui le distinzioni tra agenti umani, artificiali o ibridi, pazienti morali, tecnologie che mediano l'agire e progettisti degli artefatti diventano sempre più porose, e la sfera etica coinvolge tutti i poli di questa relazione, inscrivendo la responsabilità direttamente nei processi di design.

Una sintesi particolarmente efficace di questa metamorfosi è stata proposta da Vilém Flusser (1920–1991), filosofo ceco naturalizzato brasiliano, che già negli anni Novanta osservava come fosse divenuto estremamente arduo attribuire la responsabilità morale in una società caratterizzata dalla pervasività delle tecnologie e dalla crescente complessità della produzione industriale in via di automatizzazione.

Secondo Flusser, questo mutamento dipendeva da tre fattori principali. In primo luogo, la perdita di credibilità delle autorità tradizionali – religiose, politiche, morali – che per secoli avevano dettato norme e criteri di giudizio, ma che apparivano ormai inadeguate di fronte a processi tecnici sempre più complessi. In secondo luogo, la stessa produzione industriale e il design dei prodotti generano un ammontare di informazioni tale da eccedere le capacità cognitive individuali: la responsabilità diviene allora diluita in una cooperazione organizzativa, che coinvolge team di progettisti, produttori e persino macchine. Infine, non è più possibile considerare la responsabilità morale come imputabile esclusivamente all'utente. Se un tempo, come nel caso del coltello, l'uso improprio ricadeva solo su chi lo impugnava, oggi l'automazione dei processi produttivi e l'introduzione di artefatti sempre più autonomi rendono insostenibile

questo schema: la responsabilità deve essere ricondotta anche a chi progetta, programma e costruisce le tecnologie [Flusser, 2003, pp. 63-65].

Per chiarire fino a che punto la responsabilità possa annidarsi nei processi di progettazione, Flusser richiama un episodio drammatico emerso durante il processo di Norimberga: la lettera di un industriale tedesco che si scusava con un ufficiale nazista per aver progettato camere a gas che, difettose, “invece di uccidere migliaia di persone per volta, ne uccidevano solo centinaia” [Flusser, 2003, p. 66]. L’episodio è emblematico di come “a) non esistano più norme applicabili alla produzione industriale; b) non ci sia più un singolo autore di un crimine; e c) la responsabilità sia così diluita che di fatto ci troviamo in una situazione di totale irresponsabilità rispetto agli atti derivanti dalla produzione industriale” [Flusser, 2003, p. 66]. A questi tre punti si può aggiungere una considerazione ulteriore: il ruolo dell’artefatto stesso nel determinare un esito moralmente giudicabile. Non è più possibile concepire la relazione tra oggetto e utente come *lineare*, senza tener conto del designer e del processo di progettazione. I progettisti, dunque, non possono eludere una riflessione etica sul loro lavoro, pena il dissolvimento della responsabilità morale individuale. L’episodio richiamato mette in evidenza, in forma estrema, la funzione di mediazione esercitata dagli artefatti tecnologici nella sfera morale.

In questa direzione si colloca la riflessione di Jeroen van den Hoven (1957–), filosofo olandese della tecnologia, che ha messo in luce come sia mutata, insieme al contesto sociale e tecnologico la concezione morale tradizionale, legata al senso comune. “La tecnologia ha modificato quella che Samuel Scheffler definisce la “fenomenologia dell’agency”, così come il contesto sociale della nostra azione morale” [Van den Hoven, 2017 p.16 trad. mia]⁸⁶⁶.

La concezione morale del senso comune, citando Scheffler, per Van den Hoven implicava: “la priorità dei doveri negativi – evitare di arrecare danno agli altri – rispetto ai doveri positivi di fare il bene o di assistere gli altri nel perseguimento dei loro obiettivi”; favoriva “l’idea di una divisione del lavoro morale, secondo cui i nostri obblighi verso chi ci è più vicino – i più prossimi e cari – sono più forti rispetto a quelli nei confronti di coloro che vivono in angoli remoti della terra”; “il primato dell’atto sull’omissione” e infine “che gli effetti prossimi abbiano priorità rispetto a quelli lontani: la nostra influenza causale è percepita come inversamente proporzionale alla distanza spaziale e temporale degli effetti.” [van den Hoven,

⁸⁶⁶ “Technology has changed what Samuel Scheffler calls ‘the phenomenology of agency’ as well as the social context of our moral agency”.

2017, p.16-17 trad. mia]⁸⁶⁷. La diffusione delle tecnologie, nel senso che abbiamo delineato, rende questa posizione tradizionale insostenibile. “il mondo ampliato, in cui siamo tutti connessi tramite Internet, infrastrutture e trasporti aerei, sembra richiedere una concezione altrettanto ampliata della moralità e della responsabilità, meno restrittiva rispetto alla concezione del senso comune con cui abbiamo finora operato” [Van den Hoven, p.16 trad. mia]⁸⁶⁸

Quest’ultima non è più adeguata a descrivere il mondo contemporaneo, segnato dalla globalizzazione e dalla pervasività delle tecnologie digitali, in cui l’agency morale è distribuita tra una pluralità di attori e intrecciata con ambienti tecnologici complessi. “Secondo Scheffler, la sussunzione degli agenti individuali in strutture globali complesse e interconnesse presenta una serie di caratteristiche moralmente rilevanti: (1) il contributo degli individui ai risultati complessivi è limitato; (2) il loro controllo è ridotto; (3) i processi in cui sono inglobati sono pervasivi e inevitabili; e (4) è molto difficile ottenere informazioni accurate, affidabili e tempestive su tali processi e sui loro effetti” [Van den Hoven, 2017, p.18 trad. mia]⁸⁶⁹.

. Van den Hoven aggiunge un punto decisivo: molte delle azioni che compiamo oggi non avvengono più direttamente tra individui, ma *attraverso la tecnologia* e dentro un *milieu tecnologico progettato*, frutto di scelte accumulate nel tempo e nello spazio da altri agenti. Per questo, nella riflessione morale devono essere prese in considerazione “1) le storie di *design* dei sistemi e dell’ambiente tecnologico in cui è ambientata l’agency; e 2) le scelte e il ruolo degli

⁸⁶⁷ “The enlarged world, in which we are all connected via the Internet, infrastructures and airplanes, seems to require an equally enlarged conception of morality and responsibility that is less restrictive than the commonsense conception of morality and responsibility that we have been working with thus far. Scheffler points out that the commonsense conception of morality typically asserts the priority of negative duties to avoid harm to others over positive duties to do good or assist others in the pursuit of their goals. It furthermore favours the idea of division of moral labour, which implies that our obligations to those closer to home, the nearest and dearest, are stronger than our obligations to those in the remote corners of the earth. Third, our commonsense morality also implies a primacy of acts over omissions, since acts are always experienced as acts of one self, whereas omissions are only experienced as such in specific contexts. Typically, they are experienced only as omissions when a norm is violated as a result of not doing something. The commonsense conception furthermore implies that near effects have primacy over remote effects. We experience our causal influence as inversely related to spatial and temporal distance of the effects. We also experience our own individual agency as such to a lesser extent when we act together with others in a more complex social context.”

⁸⁶⁸ “The enlarged world, in which we are all connected via the Internet, infrastructures and airplanes, seems to require an equally enlarged conception of morality and responsibility that is less restrictive than the common sense conception of morality and responsibility that we have been working with thus far”.

⁸⁶⁹ “The subsumption of individual agents in complex global interconnected structures has a number of morally relevant features according to Scheffler: (1) the contribution of individuals to overall results is limited; (2) their control is limited; (3) the processes in which they are subsumed are pervasive and inescapable; and (4) it is very difficult to get accurate, reliable and timely information about these processes and their effects”

agenti di *design* nella formazione delle architetture di scelta” [Van den Hoven, 2017, p.19 trad. mia]⁸⁷⁰.

In qualunque ambiente informazionale – dalla sala di controllo del traffico aereo alle piattaforme digitali – le scelte di design risultano decisive: se il software contiene bug, se i modelli si fondano su presupposti errati, se i dati sono imprecisi, le decisioni e le azioni umane ereditano questi difetti. È in questo senso che si configura il *design turn* dell’etica applicata: la consapevolezza che la progettazione non sia un’attività neutrale, ma il luogo in cui i valori vengono tradotti in pratiche e in cui si definiscono le condizioni stesse della responsabilità morale [Van den Hoven, 2017, p.25-26].

"La svolta progettuale nasce dalla consapevolezza che il mondo moderno è un artefatto, popolato da numerosi manufatti". La saturazione di questi sistemi orienta l’attenzione non più su teorie morali astratte, ma sulle storie di progettazione, la loro implementazione e l’impatto sugli esseri umani. "Se Nietzsche ha tracciato la genealogia della morale, la svolta progettuale ha tracciato la genealogia delle scelte progettuali. Ciò implica che l’individuo (designer) non è l’unico portatore di responsabilità morale" [Umbrello, 2023].

Come abbiamo notato attraverso le considerazioni di Galison sul metodo Monte Carlo, “bit dopo bit (byte dopo byte), i progettisti di computer hanno decostruito la nozione stessa di strumento, via via che il computer finiva per rappresentare non più uno strumento, bensì la natura stessa”. La dimensione etica delle tecnologie digitali, coerentemente con quanto osservato a proposito della mutazione della fenomenologia morale, non può quindi restare confinata alla sola *Computer Ethics* (CE), intesa come un’“etica del carpentiere” [Floridi, 2009, p. 28]⁸⁷¹, che considera le tecnologie soltanto come strumenti valutabili secondo le categorie dell’etica classica. È necessario un ampliamento di prospettiva.

Un paradigma capace di interpretare questo scenario mutato, in linea con la nostra analisi, è l’*Information Ethics* (IE) [Floridi, 2008, p. 3]. L’IE assume “l’informazione come livello d’astrazione comune per la considerazione di ogni ente, il quale è a un livello ontologico minimale non contraddittorio e denotabile” [Floridi, 2009], elevandola così oltre i limiti

⁸⁷⁰ “Much of what we do in the twenty-first century we do not do directly to each other, but almost always by way of technology and via a technological milieu that was designed, and that came about over the course of centuries as a result of myriad contributions by others separated from us by time and place. This significantly changes our moral situation and adds to the need for a different conception of moral agency. It notably shifts the locus of responsibility away from the agent as traditionally conceived and brings into view (1) the *design histories* of the systems and technological milieu in which agency is staged and (2) the choices and role of *design agents* in the shaping of choice architectures”.

⁸⁷¹ Floridi sostiene che questa sia la posizione difesa, indirettamente da Johnson, 1985

dell'etica standard - spesso segnata da un bias antropocentrico (si pensi alle difficoltà di universalizzazione degli imperativi kantiani) - e superando, al contempo, i limiti delle etiche non-standard, come quelle biocentriche, nel riconoscere nell'*informazione* "vero soggetto paziente dell'etica" [Floridi, 2009, p. 34].

L'IE si configura dunque come "una macroetica orientata al paziente, ontocentrica ed ecologica" [Floridi, 2010, p. 83 trad. mia]⁸⁷², fornendo un quadro teorico entro il quale collocare, nell'orizzonte tecnologico contemporaneo, la riflessione sul ruolo del designer. Quest'ultimo è infatti chiamato a realizzare valori in artefatti che, lungi dall'essere meri strumenti, devono essere riconosciuti anch'essi come agenti morali, capaci di generare conseguenze eticamente rilevanti.

Il design deve tenere conto che ciò che progetta e implementa può coinvolgere entità come "specie non umane, superorganismi, la Terra, i social robot" (Friedman, Hendry, 2019, p.27)⁸⁷³. Tutti questi enti hanno una comune matrice nell'informazione, e l'IE può essere l'orizzonte concettuale per il trattamento morale non solo degli Agenti Artificiali, ma di tutto l'insieme degli enti considerati al LoA minimale dell'informazione. Enti "che possono essere persone, quartieri, comunità, organizzazioni, istituzioni o società e possono includere le passate e future generazioni, le specie non umane e altri elementi come edifici storici o cime di montagne sacre"⁸⁷⁴ [Friedman, Hendry, 2019, p.37].

Una prospettiva promettente per la materializzazione⁸⁷⁵ dei valori nelle tecnologie digitali, che prende le mosse da questa prospettiva, e compatibile con lo sviluppo dell'AI4SG [Cfr. Umbrello, 2023], è il *Value Sensitive Design* [VSD].

⁸⁷² "IE is an *ontocentric, patient-oriented, ecological* macroethics"

⁸⁷³ "The term "entity" is used here to account for nonhuman stakeholders such as an organization, nonhuman species, or sacred mountaintop"

⁸⁷⁴ "Yet the technologies we design and build reach far beyond human beings to implicate other nonhuman entities. Here are four for consideration: nonhuman" species, superorganisms, the Earth, and social robots"

⁸⁷⁵ Gli oggetti in quanto non sono neutri, operano mediando l'azione una forma di influenza più o meno indiretta di, spinta, di *nudging*: "Qualsiasi aspetto dell'architettura delle scelte che altera il comportamento delle persone in modo prevedibile senza vietare alcuna opzione o cambiare significativamente i loro incentivi economici" ["A nudge, as we will use the term, is any aspect of the choice architecture that alters people's behavior in a predictable way without forbidding any options or significantly changing their economic incentives Thaler, Sunstein, 2021, p.8 trad. mia] L'incarnazione di valori nella materialità degli oggetti, l'*ethics by design*, può essere vista come una forma di nudging ontologico, che viola l'autonomia decisionale del soggetto che non può non agire come l'oggetto impone. A questa forma L. Floridi oppone il *pro-ethical design*, una forma di nudging informazionale, modifica la natura delle informazioni a cui le persone sono esposte, influenzando così le decisioni senza alterare le opzioni fisiche disponibili, come più rispettoso dell'autonomia dell'agente [Floridi, 2016]. Per quanto riguarda la responsabilità morale dei designer in relazione ad azioni intenzionali o non-intenzionali ed effetti prevedibili o non prevedibili, si rimanda a Berdichevsky, Neuenschwander. 1999.

Nato negli anni '90 presso l'Università di Washington [Friedman et al., 2008; Friedman, Hendry, 2019; Davis, Nathan, 2015], il VSD è un approccio teoricamente fondato che integra in modo sistematico i valori umani⁸⁷⁶ in tutte le fasi della progettazione. Esso rappresenta «un modo di fare etica che mira a rendere i valori morali parte della progettazione, della ricerca e dello sviluppo tecnologico» [Van den Hoven, 2017, p. 28 trad. mia]⁸⁷⁷, orientando il design oltre la mera funzionalità tecnica e incorporando nel processo decisionale dimensioni morali quali equità, giustizia, benessere, dignità e sostenibilità ambientale.

Il VSD si fonda su una metodologia tripartita - indagini concettuali, empiriche e tecniche [Friedman et al., 2008] - applicata in modo iterativo, e considera sia gli stakeholder diretti sia quelli indiretti, ampliando così la portata etica del design⁸⁷⁸. In questo senso, il VSD si pone come strumento chiave per il “*design turn*” contemporaneo, capace di tradurre in artefatti tecnologici i valori condivisi, riducendo al contempo bias e rischi sociali, e candidandosi come paradigma compatibile con l'implementazione responsabile di sistemi di IA orientati al bene sociale [Umbrello, 2023].

Come dare corpo al *design turn* dell'etica, nella prospettiva che abbiamo delineato, all'interno degli spazi in cui si implementano materialmente le tecnologie? Nell'orizzonte del

⁸⁷⁶ Friedman e agli altri studiosi che hanno sviluppato e propugnato il VSD hanno nel tempo proposto un insieme di 12 valori che il Value Sensitive Design deve tenere in conto nella progettazione tecnologica. Questi valori sono: *Benessere umano; Proprietà e possesso; Privacy; Libertà dai bias; Usabilità universale; Fiducia; Autonomia; Consenso informato; Responsabilità; Cortesia; Sensibilità ai valori; Identità; Rispetto; Calma; Sostenibilità ambientale*. Questi rappresentano, nel VSD, il terreno etico del design [Friedman, Hendry, 2019, p.28].

⁸⁷⁷ “ [The VSD is] a way of doing ethics that aims at making moral values part of technological design, research and development”

⁸⁷⁸ Il VSD si caratterizza per otto *features* principali. È *proattivo*, in quanto mira a influenzare il design della tecnologia fin dalle fasi iniziali e lungo tutto il processo di sviluppo; *estensivo*, perché amplia l'ambito in cui emergono i valori includendo non solo il luogo di lavoro (come nel CSCW, *Computer Supported Cooperative Work*), ma anche istruzione, casa, commercio, comunità online e vita pubblica; *metodologicamente integrativo*, poiché combina in modo iterativo indagini concettuali, empiriche e tecniche; *valorialmente inclusivo*, dato che considera non soltanto i valori della cooperazione (CSCW) o della partecipazione (come nel PD, *Participatory Design*), ma anche quelli morali fondamentali come equità, giustizia, benessere e virtù, accanto a convenzioni e valori personali; *distintivo*, perché differenzia l'usabilità dai valori umani con impatto etico, riconoscendo che sistemi altamente usabili non necessariamente supportano valori morali; *stakeholder-oriented*, in quanto prende in considerazione sia gli stakeholder diretti, che interagiscono con il sistema, sia gli indiretti, che ne subiscono gli effetti; *interazionale*, nella misura in cui i valori sono compresi come frutto dell'interazione tra design e contesto sociale, evitando spiegazioni esclusivamente endogene o esogene; infine *universalista*, perché assume che alcuni valori siano condivisi trasversalmente, pur manifestandosi in forme diverse a seconda delle culture e dei contesti [Friedman et al., 2008; 2013]. Tuttavia, il VSD non è esente da criticità. Tra le principali emerge la mancanza di una chiara definizione di “valore”, insieme al rischio di oscillare tra relativismo e universalismo. In particolare, gli è stata imputata la tendenza all'*ipostatizzazione valoriale*, ossia la reificazione di un insieme ristretto di valori che rischia di non adattarsi ai contesti locali e di ostacolare l'emersione di nuovi valori. Al VSD viene inoltre attribuita la *fallacia naturalistica*, cioè la confusione tra ciò che è e ciò che dovrebbe essere. Queste problematiche evidenziano la difficoltà di bilanciare l'universalità dei valori con la loro adattabilità ai diversi contesti socio-culturali e progettuali, una questione tuttora aperta nella sua metodologia. Per le critiche al VSD, si vedano Albrechtslund, 2007; Le Dantec et al., 2009; Mander-Huits, 2011.

VSD si apre lo spazio per una figura dotata di competenze in campo etico (l'eticista o *value advocate*), cioè l'*eticista designer*.

Una svolta pragmatica dell'etica in progettazione richiede una figura integrata nel team, capace di unire competenze etiche, epistemologiche e tecniche per chiarire i valori in gioco, mediare tra prospettive e tradurre opzioni normative in scelte progettuali verificabili. In questa chiave, l'eticista agisce come *value advocate*, come sostenitore dei valori, in modo "blended, not bossy": è innestato nei flussi di lavoro, porta nel team letteratura e strumenti (storia della filosofia e della scienza, giustizia sociale, studi ambientalisti, etica professionale del design, etica della tecnologia, pratica tecnica critica); funge da traduttore tra valori astratti e *affordance* tecniche; attiva conversazioni interdisciplinari e fa emergere l'"ecologia morale" già presente in laboratorio o progetto [Shilton, Anderson, 2016].

Riguardo all'incorporare l'etica nei team di design. La tensione è tra due correnti principali: La prima converge su un approccio pragmatico, in cui tecnici (scienziati, designer, ingegneri, programmatori) integrano l'etica nelle pratiche di ricerca e progettazione - qui rientra il VSD - e l'eticista favorisce il design anziché "ostacolarlo" con un'astratta discussione sui valori; la seconda prospettiva è quella sostenuta dagli scienziati sociali, secondo cui l'eticista lavora in laboratorio accanto al team, con lo scopo di aprire spazi di discussione in cui designer e tecnici possano identificare valori, dilemmi e priorità [van Wynsberghe, Robbins, 2013].

Per superare questa dicotomia, viene proposto un "ruolo innovativo per l'eticista: l'eticista come progettista, che adotta una visione pragmatica dell'etica per introdurre l'etica nella ricerca e nella progettazione di artefatti, indipendentemente dallo stadio di sviluppo» [van Wynsberghe, Robbins, 2013 trad. mia]⁸⁷⁹.

Questa impostazione riconosce che il design - soprattutto di artefatti tecnologici e digitali - coinvolge molteplici attori portatori di competenze e valori differenti, da coordinare razionalmente in vista di scelte progettuali: "Ci sono manager aziendali che comprendono e difendono il valore economico, ci sono ingegneri che comprendono e difendono i valori ingegneristici (ad es. efficienza e velocità), ci sono marketer e designer grafici che difendono i

⁸⁷⁹ "It is our contention that ethics ought to be pragmatic and to provide utility for the design process and we maintain that adequate ethical reflection, and all that it entails, ought to be conducted by an ethicist. Thus, we propose a novel role for the ethicist-the ethicist as designer-who subscribes to a pragmatic view of ethics in order to bring ethics into the research and design of artifacts-no matter the stage of development"

valori estetici e, infine, ci dovrebbero essere eticisti che difendono i valori etici e morali in modo competente” [van Wynsberghe, Robbins, 2013 trad. mia]⁸⁸⁰

Operativamente, l’eticista designer svolge un ruolo di traduzione dei valori in concrete caratteristiche dell’artefatto: a partire dalla scoperta dei valori (*value discovery*), rende espliciti i valori rilevanti di stakeholder diretti e indiretti, norme professionali, standard e leggi, mettendo a fuoco priorità e trade-off; passa quindi alla concettualizzazione e allo scrutinio, chiarendo il significato operativo dei valori rispetto a dominio, contesto d’uso e attese del progetto; approda infine alla traduzione in progetto (*value translation/specification*), trasformando i valori in norme orientative e in requisiti di design testabili che guidano decisioni, metriche e verifiche lungo lo sviluppo [van Wynsberghe, Robbins, 2013]

Questa scansione - *scoperta* → *concettualizzazione/scrutinio* → *traduzione* - definisce il profilo dell’eticista come designer e ne istituisce il ruolo come membro stabile del team, pragmatico e utile al progetto.

In questa prospettiva, la traduzione è la fase cruciale del VSD: il ponte che collega i valori alle *affordance* effettive dell’artefatto. Qui l’eticista designer media tra intenzioni normative e soluzioni tecniche, assicurando che i valori diventino specifiche implementabili. Seguendo la traccia proposta da Van de Poel [Van de Poel, 2013], il lavoro scorre dal livello dei valori a quello delle norme - includendo leggi, policy aziendali, codici professionali e standard tecnici - fino ai requisiti di progettazione. La traduzione procede attraverso due movimenti intrecciati, *bottom-up* (dai valori alle norme) e *top-down* (dalle norme ai requisiti). Non è una deduzione meccanica: i livelli inferiori sono più concreti e dipendono dal contesto; servono giudizi di valore e conoscenza del dominio per trasformare norme e policy in requisiti verificabili e, in ultima istanza, in concrete *affordance* progettuali. [Van de Poel, 2013].

Dalla prospettiva che abbiamo analizzato-che, per le tecnologie digitali in tutte le loro forme fino all’IA contemporanea, sposta l’attenzione sul piano dell’azione, dell’etica e della loro traduzione operativa-emerge una tendenza che prende corpo nella figura professionale capace di farne pratica: l’eticista designer.

In continuità con le nostre analisi sulla cibernetica e sulla teoria dell’informazione, questa traiettoria coincide con un *blurring* dei confini concettuali che mette in questione le

⁸⁸⁰“Design is a process achieved by a team of people. There are business managers understanding and advocating for economic value, there are engineers understanding and advocating for engineering values (e.g., efficiency and speed), there are marketers and graphic designers advocating for aesthetic values, and finally there should be ethicists advocating for ethical and moral values in a skilled manner. Below we conceptualize the ethicist as a member of the design team”.

separazioni tra soggetto e ambiente, individuo e cultura [Cfr. Boden, 2006, p. 1038]. A partire da questo orizzonte, il compito è favorire la *comunicazione* tra i diversi protagonisti-umani, artificiali e ibridi-e assicurare il *controllo* umano sulle nuove tecnologie, così che l'etica non si riduca a mero *bluwashing* ma diventi infrastruttura di progetto: valori resi *affordance* e requisiti verificabili, orientati a fare dei continui sviluppi tecnologici un vettore di *innovazione sostenibile*, rispettosa dei valori e soggetta al controllo democratico.

DIGITAL DESIGN: Un framework etico per la progettazione di tecnologie AI-HCI come soluzione sostenibile alle sfide della complessità e dell'innovazione

Il seguente elaborato è stato realizzato nell'ambito della componente aziendale del presente progetto di ricerca, sviluppata presso Exprivia S.p.A., all'interno del team di *design thinking* della DFXM. L'attività progettuale ha costituito un momento di sperimentazione applicativa dei principi teorici discussi, consentendo di tradurli in un artefatto operativo di supporto alla progettazione di tecnologie AI-HCI. Si ringraziano il co-tutor dott. Vitantonio Fascilla, il dott. Nicola Tino, la dott.ssa Tiziana Giove e il dott. Domenico Favuzzi per il supporto scientifico, metodologico e organizzativo fornito nel corso delle attività e la preziosa opportunità offertami

Digital Design

**Un framework etico
per la progettazione
di tecnologie AI-HCI
come soluzione sostenibile
alle sfide della complessità
e dell'innovazione**



INDICE DEI CONTENUTI

1. DESIGN E ORGANIZZAZIONE DELL'INFORMAZIONE

- DUE ANIME RIUNITE NEL DESIGN: COGNITIVA E INGEGNERISTICA
- INTERAZIONE
- FONDAMENTI: L'UOMO AL CENTRO
- ONTOLOGIA DEI DATI/BIG DATA
- ENTROPIA - INFORMAZIONE SEMANTICA/SINTATTICA
- FONDAMENTI: STRUTTURE E REGOLE -
- FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

2. DESIGN AI

- MACHINE LEARNING
 - Machine learning
 - Induzione
- DEEP LEARNING
 - Deep learning
 - Connessionismo
 - Limiti del deep learning
- LARGE LANGUAGE MODELS
 - Large language models
 - Rischi e criticità degli LLMs
 - Dati e addestramento: rischi e costi
 - Pappagalli Stocastici
- KNOWLEDGE CONDITION E COORDINATION NORMS
- ARTIFICIAL INTELLIGENCE FOR SOCIAL GOOD (AI4SG)- SUSTAINABLE DEVELOPMMENT GOALS
 - Bias, classificazione, discriminazione
 - Loop algoritmico discriminatorio
- MAPPING ETICO ALGORITMI
- FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE
 - 1 Falsificabilità e implementazione incrementale
 - 2 Garanzie contro la manipolazione dei predittorii
 - 3 Intervento contestualizzato in favore del destinatario
 - 4 Spiegazione contestualizzata in favore del destinatario e finalità trasparenti-
Esempi-*Livelli di Astrazione-*LoA e Ontologia: essere presenti- *LoA e agency morale degli agenti artificiali
 - 5 Tutela della Privacy e consenso dell'Interessato
 - 6 Equità concreta
 - 7 Semantizzazione adatta all'umano
- MATRICI ETICHE
 - Matrice etica
 - Matrice Etica: Principi e Stakeholder
 - Matrice Etica: Principi e Stakeholder: Mock up visualizzazione grafica

3. OVERLAPPING ISSUES/FEATURES ETICHE AI(ML)-HCI

- ISSUES/FEATURS ETICHE
 - Privacy
 - Privacy by Design
 - Tracciabilità: Base per l'attribuzione di responsabilità nei sistemi AI(ML)-HCI

4. DESIGN HCI

- LE TRE WAVE DELL'HCI
- EVOLUZIONE DELL'HCI: DAL MAINFRAME ALL'UBIQUITÀ...E OLTRE

- HCI: ontogenesi del soggetto
- Interfacce umane
- Tassonomia delle Interfacce/Esempi di valorizzazione e rischi etici
 - PERSUASIONE E NUDGING
- Persuasione
- Dark Pattern
- La spinta gentile: verso il pro-ethical design
 - MAPPATURA ETICA DELLE EURISTICHE PER IL DESIGN
- 1 Visibilità dello stato del sistema
- 2 Corrispondenza tra sistema e mondo reale
- 3 Controllo dell'utente e libertà
- 4 Coerenza e standard-Esempi-*Cognizione-*Esempi di tecniche e consigli di design basatl sulle attività cognitive della mente
- 5 Prevenzione degli errori- *Modelli mentali
- 6 Riconoscimento piuttosto che ricorso
- 7 Flessibilità ed efficienza d'uso
- 8 Design estetico minimalista
- 9 Aiutare gli utenti a riconoscere, diagnosticare e recuperare dagli errori -Esempi
- 10 Aiuto e comunicazione -
- Correlazione Principi Etici/Euristiche del Design

5. DESIGN PER UNA SOCIETÀ EQUA

- DESIGN PER UNA SOCIETÀ EQUA: AMBIENTE E INCLUSIVITÀ

- Green
 - Green Coding
 - Linguaggi di programmazione ed efficienza energetica
 - Design HCI green: Esempi di best practices per interfacce utente eco-sostenibili

Inclusività: il design per l'equità concreta

- Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze di genere
- Best Practices ed esempi concreti per il design AI e la Data Science volto a ridurre le disuguaglianze di genere
- Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze basate sull'etnia
- Best Practices ed esempi concreti per il design algoritmico volto a ridurre le disuguaglianze basate sull'etnia

6. INCORPORARE L'INNOVAZIONE SOSTENIBILE NELLE COSE: IL DESIGN THINKING E L'ETICISTA DESIGNER

- DESIGN THINKING
 - Discovery
 - Esempi di cattive pratiche nella fase di Discovery
 - Esempi di buone pratiche nella fase di Discovery
 - User Personas: rischi e strategie di mitigazione i
 - Prototipation
 - *Epistemologia Falsificazionista
 - INTRODURRE L'ETICA NEI TEAM DI DESIGN: L'ETICISTA DESIGNER
 - L'eticista designer
 - L'eticista designer nel team: il contributo alla traduzione dei valori in requisiti di progettazione - Esempi

CONCLUSIONE: FROM TOOL TO.....TOOL DALLA PAURA ALLA SPERANZA

1. DESIGN E ORGANIZZAZIONE DELL'INFORMAZIONE

- DUE ANIME RIUNITE NEL DESIGN: COGNITIVA E INGEGNERISTICA
- INTERAZIONE
- FONDAMENTI: L'UOMO AL CENTRO
- ONTOLOGIA DEI DATI/BIG DATA
- ENTROPIA - INFORMAZIONE SEMANTICA/SINTATTICA
-
- FONDAMENTI: STRUTTURE E REGOLE
- FATTORI E BEST PRACTICES PER IL DESIGN E L'IMPLEMENTAZIONE
- PRINCIPI ETICI AI/HCI

DUE ANIME RIUNITE NEL DESIGN: COGNITIVA E INGEGNERISTICA

- **COSTRUZIONE DI ARTEFATTI “COGNITIVI” = CAPACI DI MIGLIORARE LA CAPACITÀ UMANA DI INTERFACCIARSI CON LA COMPLESSITÀ SEMPRE CRESCENTI DI INFORMAZIONE**

- ANTECEDENTI STORICI NELLA PSICOLOGIA DELL'ORGANIZZAZIONE DI INIZIO '900 (TAYLOR, GILBRAITH), NELLE VISIONI DEI PIONIERI DELLE TECNOLOGIE INFORMATICHE (COME IL MEMEX DI BUSH, L'AUGMENTING HUMAN-INTELLECT DI ENGELBART, LA SIMBIOSI UOMO COMPUTER DI LICKLIDER, L'HYPertext DI NELSON) E CIBERNETICHE

HCI

- **RISERVA E AMPLIFICAZIONE INGEGNERIZZATA DELLA CAPACITÀ UMANA DI AGIRE**

- SETTORE TECNOLOGICO CHE MIRA A PRODURRE STRUMENTI CHE AUMENTANO LA CAPACITÀ DI “AGIRE A PORTATA DI MANO” DELL'ESSERE UMANO. L'AI COME SETTORE DELLE NEUROSCIENZE/SCIENZE COGNITIVE MIRANTE A RIPRODURRE L'INTELLIGENZA UMANA: FALLIMENTO

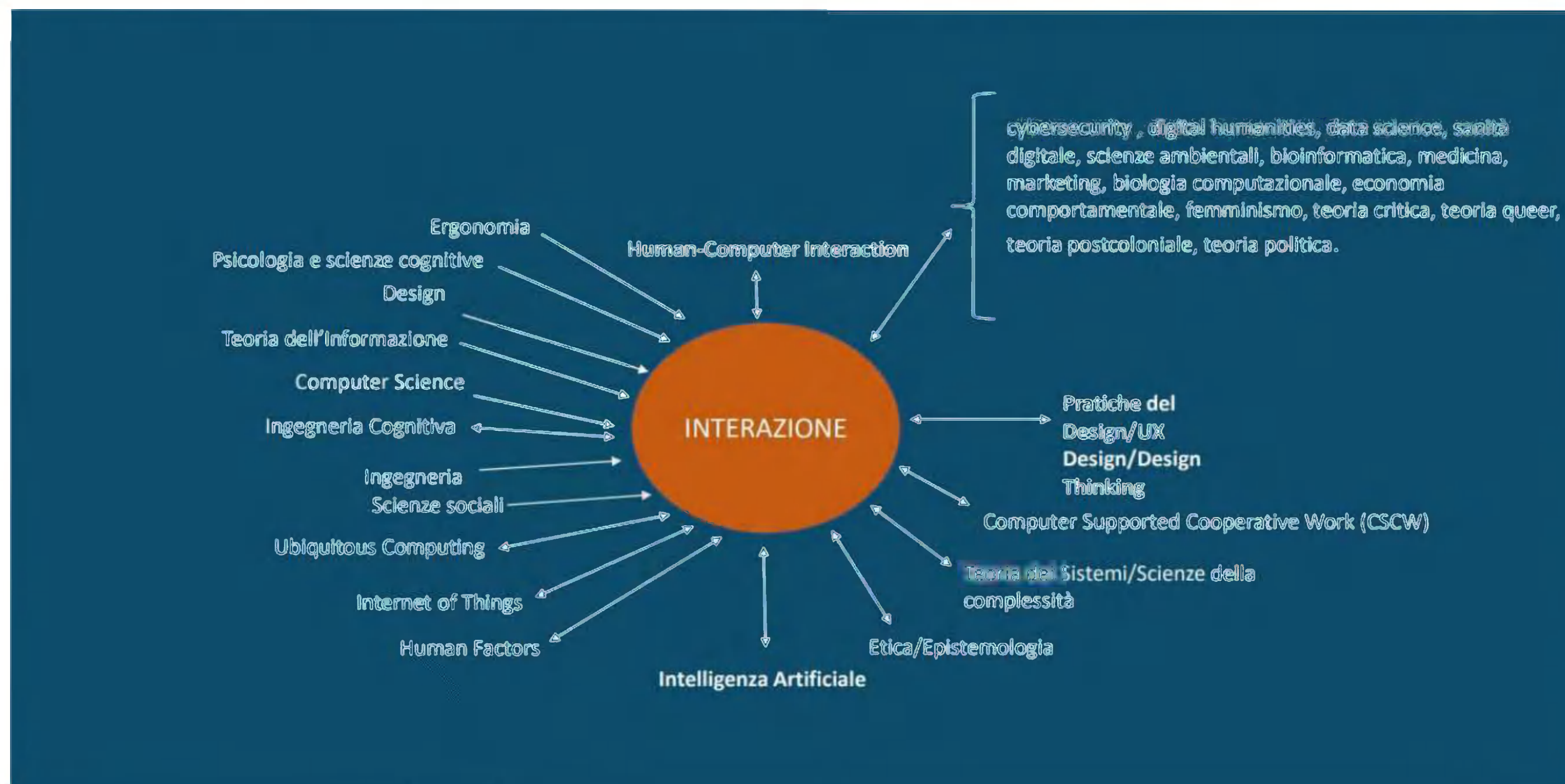
AI

Nella contemporaneità tecnologica caratterizzata da: cognitività diffusa nel mondo (l'esocorteccia della “mente estesa”), da ambienti sempre più avvolti dalla tecnologia e dalla mediazione etica degli artefatti →

Il **DESIGN come scienza dell'artificiale eticamente connotata** diviene - gestendo **vincoli** e **affordances** rivoluzionati dal digitale e progettando il modo e gli strumenti di **interazione**, **azione** e **creazione** del sè in relazione all'ambiente informazionale - il modo di riunire le due anime, **cognitiva** e **ingegneristica**, delle tecnologie e promuoverne un uso al servizio dell'essere umano e delle sfide del nostro tempo

INTERAZIONE

“Progettare spazi per la comunicazione e l’interazione umana” con esperienze e prodotti interattivi per supportare il modo in cui le persone comunicano e interagiscono nella loro vita quotidiana e lavorativa, in un ambiente che perde via via la distinzione tra online e offline e che si popola sempre più di enti artificiali e ibridi.



FONDAMENTI: L'UOMO AL CENTRO

HUMAN COMPUTER INTERACTION (HCI)

Disciplina che si occupa della progettazione, valutazione e implementazione di sistemi informatici interattivi per uso *umano* e dello studio dei principali fenomeni che li circondano

ALGORITMO: UNA DEFINIZIONE PRAGMATICA

Un algoritmo è una struttura di controllo composta finita, astratta, efficace, imperativamente data, che realizza uno *scopo dato sotto condizioni date*.

-
- LA CENTRALITÀ DELL'ESSERE UMANO È NEL CUORE DELLE DEFINIZIONI FONDAMENTALI PER IL *DESIGN* E L'IMPLEMENTAZIONE TECNOLOGICA E DIGITALE
 - LA DIMENSIONE ETICA, COME L'UOMO AGISCE CON, SULLA E ATTRAVERSO LA MACCHINA, DIVIENE LA CHIAVE PER LA GESTIONE DELLA COMPLESSITÀ INFORMATICA
 - *ETICA* E *BUSINESS* NON SONO IN CONTRAPPOSIZIONE MA ESSENZIALMENTE CONNESSI: UNA INTERAZIONE "UMANA" CON LA TECNOLOGIA È IL FATTORE DECISIVO PER L'INNOVAZIONE SOSTENIBILE (come dettato dalle normative EU come il GDPR e nazionali)

ONTOLOGIA DATI

Dati primari: i dati principali memorizzati, progettati per essere trasmessi come informazioni

Dati secondari: l'opposto dei dati primari, costituiti dall'assenza di questi

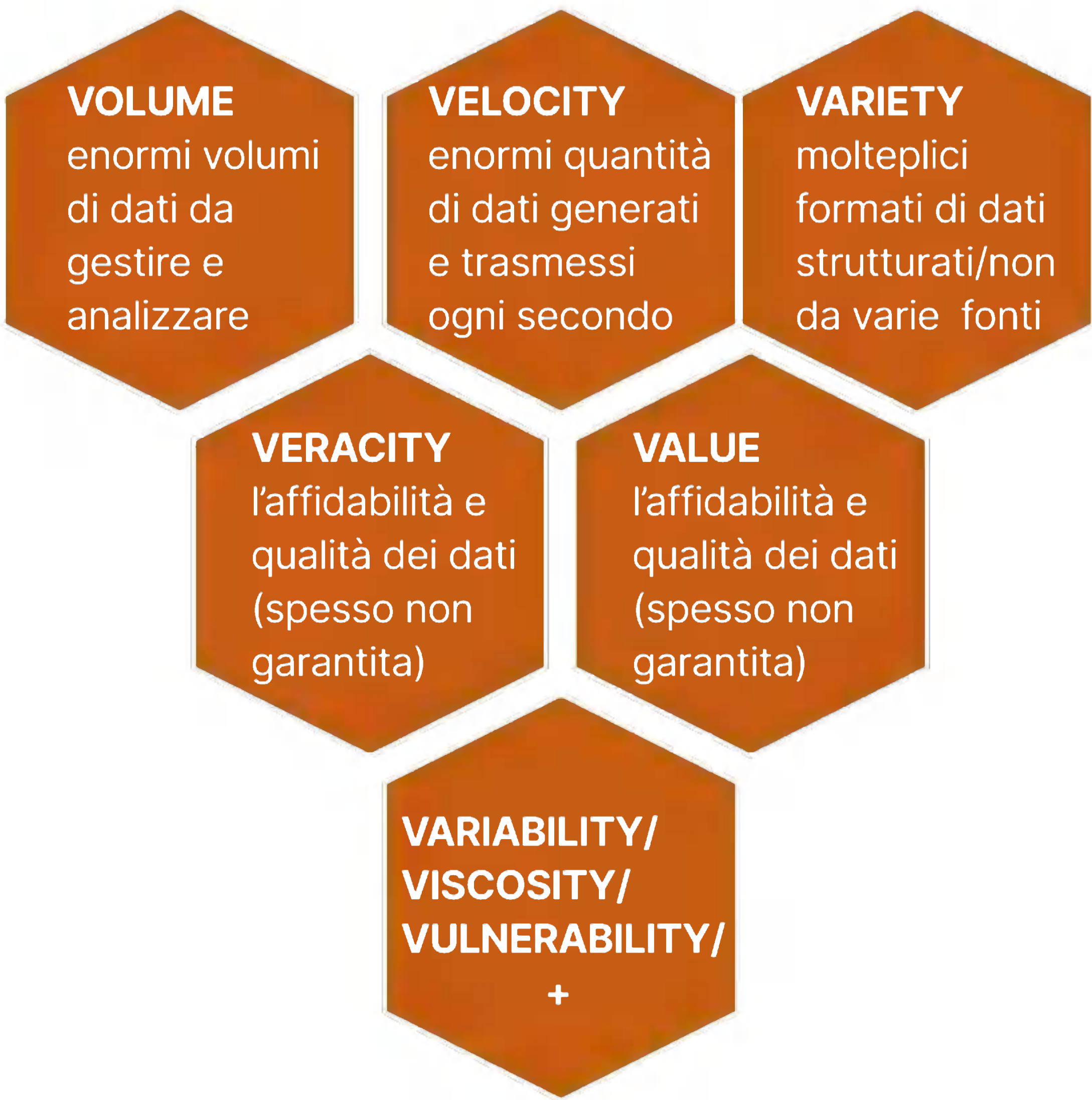
Metadati: dati sulla natura di altri dati

Dati operativi: dati riguardanti le operazioni, la struttura e le prestazioni di un intero sistema di dati

Dati derivati/Proxy: dati estratti da altre fonti e usati per inferire informazioni su cose relative ai dati originali

DATI STORICI	risultato di osservazioni sul comportamento di un sistema	regole di registrazione
DATI SINTETIZZATI	eliminano, mascherano offuscano gradi di risoluzione dei dati storici	regole di astrazione
DATI IBRIDI/ SINTETICI	generati tramite procedure	regole costitutive/ vincolanti

BIG DATA



ENTROPIA - INFORMAZIONE SEMANTICA/SINTATTICA*

L'Informazione (Shannon) si definisce in termini di distribuzione di uno spazio di probabilità e in relazione all'entropia

L'entropia quantifica l'ammontare medio di informazione prodotta da una fonte di dati probabilistica. Più alta è l'entropia, maggiore è l'incertezza associata alla fonte di informazione: l'informazione di Shannon è definita solo su termini quantitativi, senza nessun riferimento alla dimensione semantica

$$H = - \sum p_j \log_2 p_j$$

Non esistono dati puri (il puro dato è una *fictio* epistemologica) . La quantità enorme e sempre crescente di dati odierna per rappresentare informazione deve essere rappresentata epistemicamente (o implementata fisicamente)

SDI (STANDARD DEFINITION OF INFORMATION)

- **DOS** (declarative, objective, semantic);
- **σ** (infor) = simbolo usato per riferirsi a elementi discreti di informazione, a prescindere dal loro codice semiotico e dalla loro implementazione fisica;
- **δ** = dati aleticamente non qualificati, ben formati e dotati di significato:

L'informazione semantica in senso forte (RSDI) sarà così definita:

RSDI) σ è un'istanza di informazione DOS se e solo se:

1. σ CONSISTE DI N DATI (**D**), PER $N \geq 1$;
2. I DATI SONO BEN FORMATI (**WFD**);
3. I WFD SONO SIGNIFICATIVI (**MWFD = δ**);
4. I **δ** SONO **VERITIERI**.

L'ORGANIZZAZIONE/SEMANTIZZAZIONE DELL'INFORMAZIONE È IL FONDAMENTO ETICO PER LE TECNOLOGIE DIGITALI

*2011 The Philosophy of Information, Oxford University Press, Oxford; trad. it. parz, La filosofia dell'informazione, Cortina, Milano, 2024.; 1948 A Mathematical Theory of Communication, The Bell System Technical Journal, Vol. 27, pp. 379-423, 623-656; trad. it. La teoria matematica delle comunicazioni, Mimesis, Milano, 2024

FONDAMENTI

STRUTTURE

ogni ente, compreso l'essere umano, è un pattern strutturato e ordinato di informazione
il "male" etico informativo è l'entropia, il "bene" è l'integrità, organizzazione dell'informazione

- LA MIGLIORE ORGANIZZAZIONE NEGENTROPICA DELL'INFORMAZIONE MIGLIORA IL LIVELLO ETICO DELL'ARTEFATTO
-
- L'ETICA, FAVORENDO UNA GESTIONE MIGLIORE DELL'INFORMAZIONE, A LIVELLO FONDAMENTALE AUMENTA, L'EFFICACIA DELL'INTERAZIONE E DELLA PERFORMANCE
- LA COMPLESSITÀ NELL'EPOCA DEI BIG DATA È TALE CHE DAL MACROLIVELLO ONTICO DELLA TOTALITÀ ECOLOGICA DEGLI ENTI, AL MICROLIVELLO AZIENDALE = L'ORGANIZZAZIONE DELL'INFORMAZIONE E LA RIDUZIONE DELL'ENTROPIA DIVENTANO UN'AZIONE ETICA NECESSARIA DI CURA DELL'AMBIENTE INFORMATIZIONE E DELLE ENTITÀ UMANE, BIOLOGICHE, ARTIFICIALI E IBRIDE CHE LO POPOLANO

REGOLE

GROUND RULE

0- NON SI DEVE GENERARE ENTROPIA NELL'INFOSFERA

REGOLA 2

2- SI DEVE RIMUOVERE L'ENTROPIA NELL'INFOSFERA

INFOSFERA lo spazio semantico costituito dalla totalità dei dati, informazioni, conoscenze, degli agenti e delle loro operazioni

REGOLA 1

1- SI DEVE PREVENIRE L'ENTROPIA NELL'INFOSFERA

REGOLA 3

3- LO SVILUPPO DEGLI ENTI INFORMATIZIONE COME DELL'INTERA INFOSFERA DEVE ESSERE PROMOSSO, PRESERVANDO, COLTIVANDO E ARRICCHENDO LE LORO PROPRIETÀ

La **complessità** informativa odierna supera le capacità epistemiche umane

Le **Information and Communication Technologies** (ICT) permettono di accedere, semantizzare e trasformare in informazione una mole di dati altrimenti inaccessibile

Il **Design etico** di queste tecnologie è imprescindibile affinché queste tecnologie vengano progettate allo scopo di perseguire il **bene comune** e un'**innovazione sostenibile** e per far sì che siano accessibili a ognuno in modo da permettere un **enhancement** individuale verso le sfide della contemporaneità in cui l'azione mediata dalla tecnologia è imprescindibile

PRINCIPI ETICI*

I principi etici devono strutturare il design e l'implementazione sia della dimensione AI sia quella HCI e delle interfacce
L'etica riguarda le tecnologie informazionali con cui interagiamo che le modalità di questa interazione e diviene la chiave per l'innovazione sostenibile



1. BENEFICIENZA

"Fai solo il bene": promuovere il benessere, preservare la dignità e sostenere il pianeta (Tecnologie create per il benessere dell'umanità, si aggiunge la dimensione dell'implementazione del benessere individuale e della sostenibilità ambientale)



2. NON MALEFICENZA

"Non fare il male": Distinto logicamente dal "Fa' soltanto del bene" proprio della beneficenza = impedire uso improprio, stabilire un limite sicuro di sviluppo e uso. "Impedire la maleficenza del Dr. Frankenstein (più di quella del suo mostro)". Include la tutela della privacy.



3. AUTONOMIA

Evitare che autonomia artificiale mini la capacità decisionale umana; non compromettere capacità umana di darsi norme e standard; evitare che la capacità umana di ingannare/danneggiare/distruggere sia trasferibile alle macchine; rendere autonomia macchine limitata e reversibile; Include limitazioni al potere di delega alle macchine e garanzia per l'umano di poter decidere se e come decidere di nuovo



4. GIUSTIZIA

Promuovere la prosperità, preservare la solidarietà, evitare l'iniquità



5. ESPLICABILITÀ

Tiene conto della frattura tra la minoranza che sviluppa le nuove tecnologie e il resto della popolazione occidentale/mondiale la cui vita ne è modificata: Va inteso nel duplice senso:
 epistemologico: come intellegibilità= risposta alla domanda: "Come funziona?"
 etico: come responsabilità (accountability)= risposta alla domanda: "Chi è responsabile del modo in cui funziona"

2. DESIGN AI

- **MACHINE LEARNING**

- Machine learning
- Induzione

- **DEEP LEARNING**

- Deep learning
- Connessionismo
- Limiti del deep learning

- **LARGE LANGUAGE MODELS**

- Large language models
- Rischi e criticità degli LLMs
- Dati e addestramento: rischi e costi
- Pappagalli Stocastici

- **KNOWLEDGE CONDITION E COORDINATION NORMS**

- **ARTIFICIAL INTELLIGENCE FOR SOCIAL GOOD (AI4SG)**

- Bias, classificazione, discriminazione
- Loop algoritmico discriminatorio

- **MAPPING ETICO ALGORITMI**

- **FATTORI E BEST PRACTICES PER IL DESIGN E L'IMPLEMENTAZIONE**

- 1 Falsificabilità e implementazione incrementale
- 2 Garanzie contro la manipolazione dei predittori-
- 3 Intervento contestualizzato in favore del destinatario
- 4 Spiegazione contestualizzata in favore del destinatario e finalità trasparenti-

Esempi-*Livelli di Astrazione-*LoA e Ontologia: essere presenti- *LoA e agency morale degli agenti artificiali

- 5 Tutela della Privacy e consenso dell'Interessato
- 6 Equità concreta
- 7 Semantizzazione adatta all'umano

- **MATRICI ETICHE**

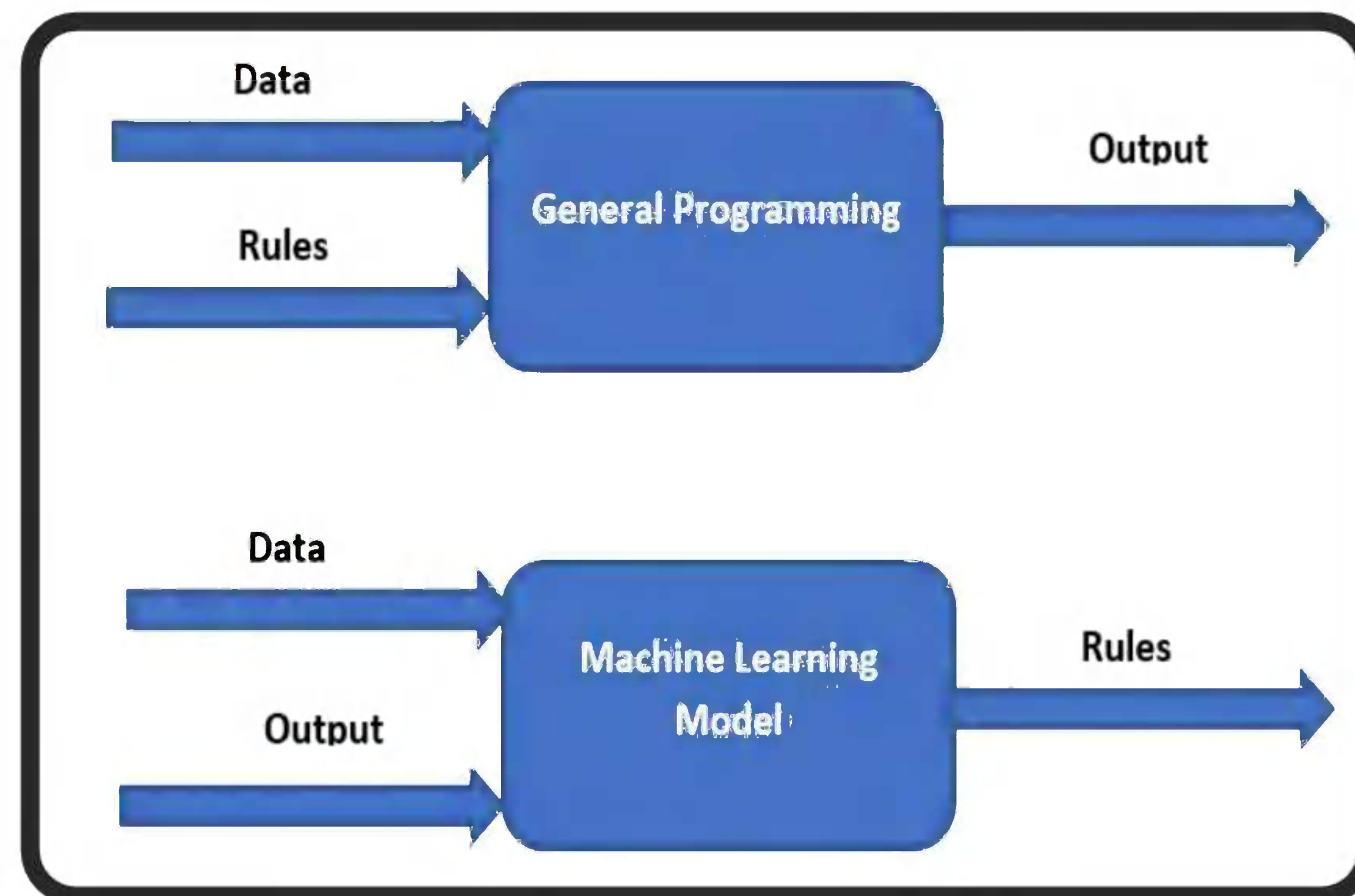
- Matrice etica
- Matrice Etica: Principi e Stakeholder: esempio
- Matrice Etica: Principi e Stakeholder: Mock up visualizzazione grafica

MACHINE LEARNING

MACHINE LEARNING

Insieme di metodi computazionali e statistici che adotta l'esperienza per il miglioramento delle prestazioni nell'individuazione di regolarità nei dati e nelle previsioni di fenomeni in scenari non ancora osservati. Gli algoritmi di ML trasformano automaticamente i dati in informazioni e in ipotesi riconoscendo delle regolarità (patterns) nei dati mediante una logica induttiva*.

Output più accurati dipendono da quantità enormi di dati (Big Data).



*Induzione

INDUZIONE

Inferenza che cerca di stabilire da casi particolari, leggi universali

Opera un aumento probabilistico della conoscenza in base ai casi già esperiti. Non giunge alla certezza

Le correlazioni inferite non sono garantite da nulla al di là della probabilità. Tralascia la conoscenza delle cause e la dinamicità della realtà in cui avvengono le osservazioni, che sono sempre parziali. L'accuratezza delle generalizzazioni dipende dalla quantità di osservazioni

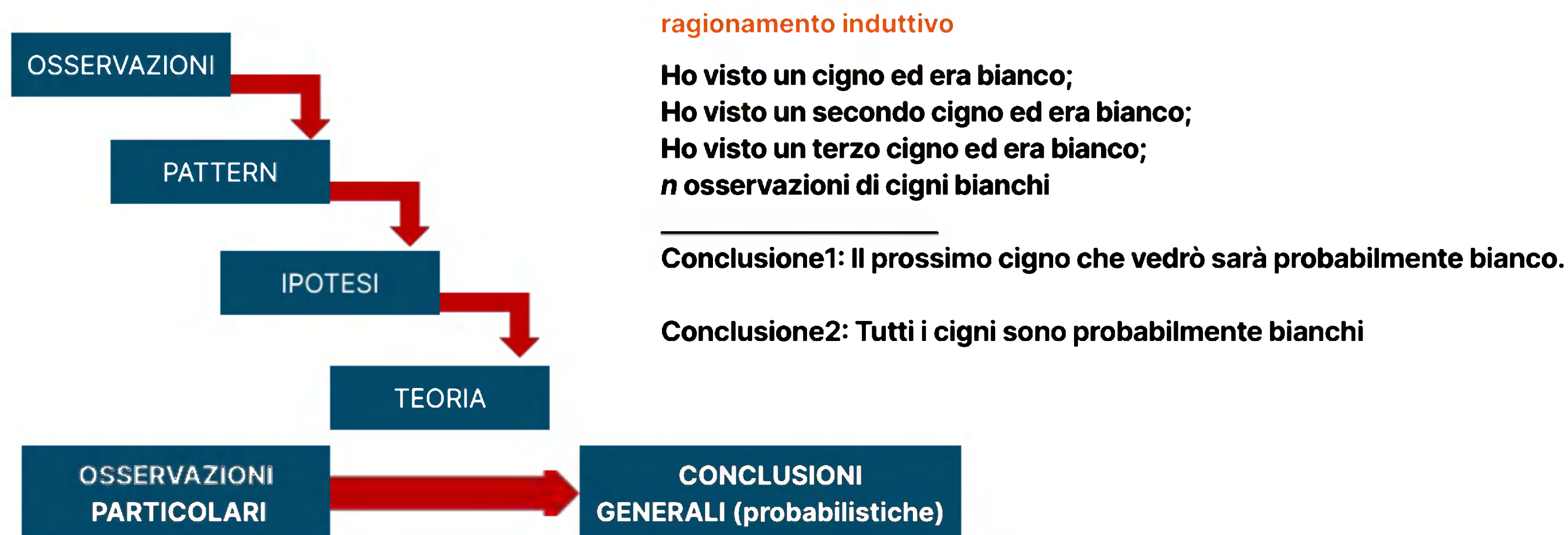
I Pattern induttivi producono una supremazia del passato sul futuro

ESEMPI

N cigni osservati sono bianchi. Quindi, Tutti i cigni sono bianchi

La percentuale Q di un campione della popolazione ha la proprietà P. Quindi, La percentuale Q della popolazione ha la proprietà P

Io sono un tacchino e mi hanno nutrito all'ingrasso dal 1 gennaio al 24 dicembre, Oggi è Natale



DEEP LEARNING

Deep learning è una sottocategoria dell'apprendimento automatico (machine learning) che si ispira alle reti intricate di neuroni del nostro cervello (e al connessionismo* neuroscientifico e filosofico), creando strati di software di Artificial Neural Network (ANN)

Ogni strato calcola i valori per quello successivo affinché l'informazione venga elaborata in maniera sempre più completa. Le reti neurali profonde (ANN), sono composte da molteplici livelli di neuroni artificiali che elaborano i dati in modo gerarchico, imparando rappresentazioni di livello sempre più astratto. Questo permette al sistema di apprendere caratteristiche complesse direttamente dai dati, senza bisogno di intervento umano.

PESO = il peso della connessione influenza la quantità di INPUT che è passato da un neurone all'altro. I pesi sono valori che vengono ottimizzati durante il processo di addestramento della rete neurale.

Segue la formula $inputs \times weights$

BIAS = a questo fattore si aggiunge il BIAS, un valore costante che va sommato al calcolo che coinvolge il peso menzionato. Anche il bias è un fattore che viene modificato durante l'addestramento.

Segue la formula $output = inputs \times weights + bias$

FUNZIONE DI ATTIVAZIONE: Dopo aver calcolato la somma pesata degli input e aggiunto il bias, il risultato può essere passato attraverso una funzione di attivazione non lineare, come la ReLU (Rectified Linear Unit), la sigmoide o la tangente iperbolica. Questa funzione determina l'output finale del neurone e introduce non linearità nel modello.

BLACK BOX = ciò che succede tra input ed output è impossibile da stabilire = non sappiamo come tutti i singoli neuroni/ANN lavorino insieme per arrivare all'output finale. Molte volte non è nemmeno chiaro cosa stia facendo un particolare neurone/ANN da solo

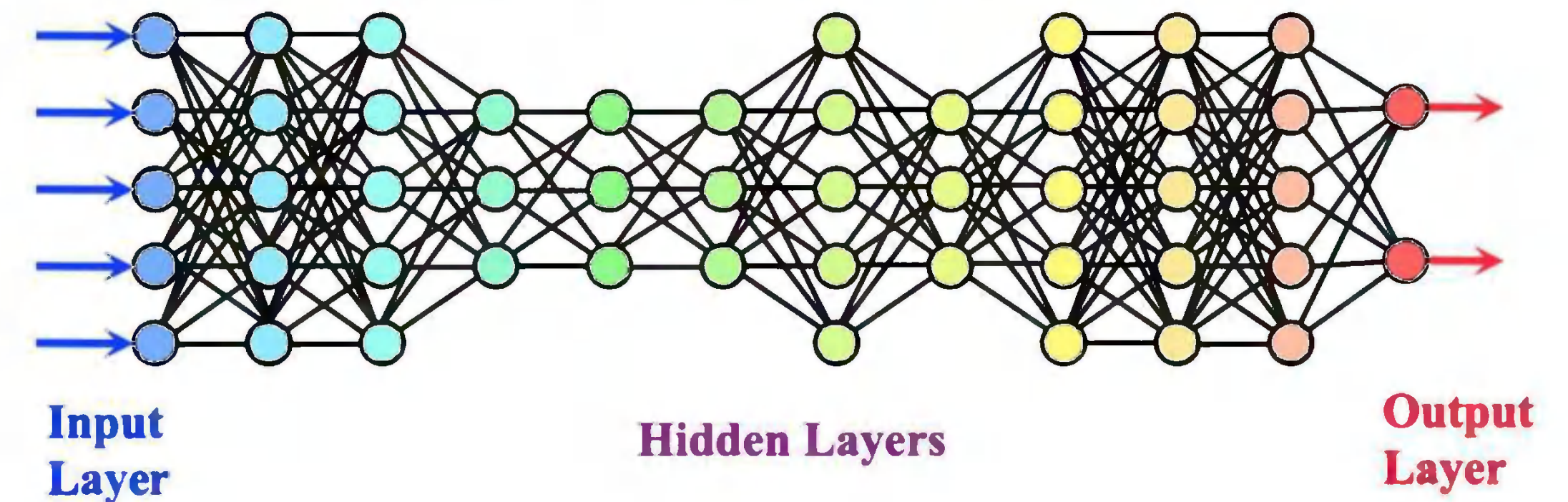
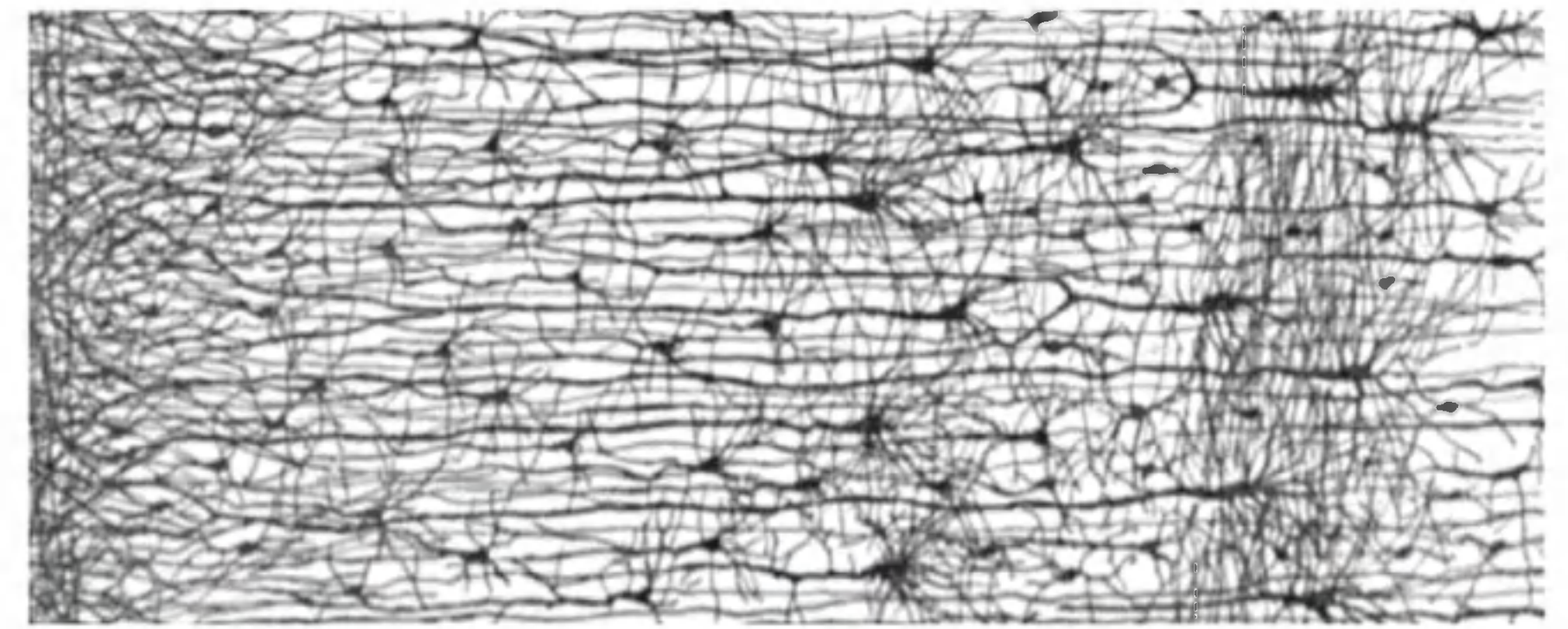


Immagine superiore: colonne di neuroni nella corteccia cerebrale disegnate dal premio Nobel del 1906 Santiago Ramón y Cajal
Immagine inferiore: una rete neurale artificiale completamente connessa con più strati nascosti, che ricorda grossolanamente il campione biologico

Processo di Calcolo

Moltiplicazione degli Inputs per i Pesi: Ogni input viene moltiplicato per il suo peso corrispondente.

$$weighted\_sum = \sum (x_i \times w_i)$$

Aggiunta del Bias: Al totale ponderato si aggiunge il bias.

$$z = weighted\_sum + b$$

Applicazione della Funzione di Attivazione: Il risultato z viene passato attraverso la funzione di attivazione per produrre l'output finale del neurone

$$output = f(z)$$

*Connessionismo

CONNESSIONISMO è un approccio in filosofia della mente e nelle neuroscienze che utilizza come modello esplicativo le reti neurali che operano in parallelo per processare l'informazione.

A differenza dell'approccio simbolico classico, che concepisce la mente come un sistema di elaborazione seriale di simboli formali, il connessionismo ritiene che i processi mentali siano emergenti dalle interazioni tra numerose unità semplici subsimboliche interconnesse.

Il connessionismo alla base dei sistemi di Deep Learning contemporanei in realtà trova i suoi antecedenti teorici nel Percettrone (Perceptron) teorizzato nel 1958 dallo psicologo USA Frank Rosenblatt (in realtà si potrebbe risalire sino agli anni 40 al fondamentale paper *A Logical Calculus of the Ideas Immanent in Nervous Activity* di McCulloch e Pitts). Questo approccio, a riprova dello sviluppo non lineare e progressivo della tecnologia, è stato bloccato dalle critiche mossegli contro nel 1969 da mostri sacri dell'AI come Marvin Minsky e Papert che hanno frenato gli investimenti in questo campo

L'approccio connessionista ha vissuto una seconda ondata nella seconda metà degli anni '80 dopo il lavoro sulle reti neurali in parallelo PDP (*Parallel Distributed Processing*) di McClelland, Rumelhart.

La contemporaneità del c. ha forse il suo emblema nella vittoria dell'algoritmo di deep learning AlphaGo che ha battuto il campione del mondo dell'antico gioco cinese Go Lee Sedol il 2016

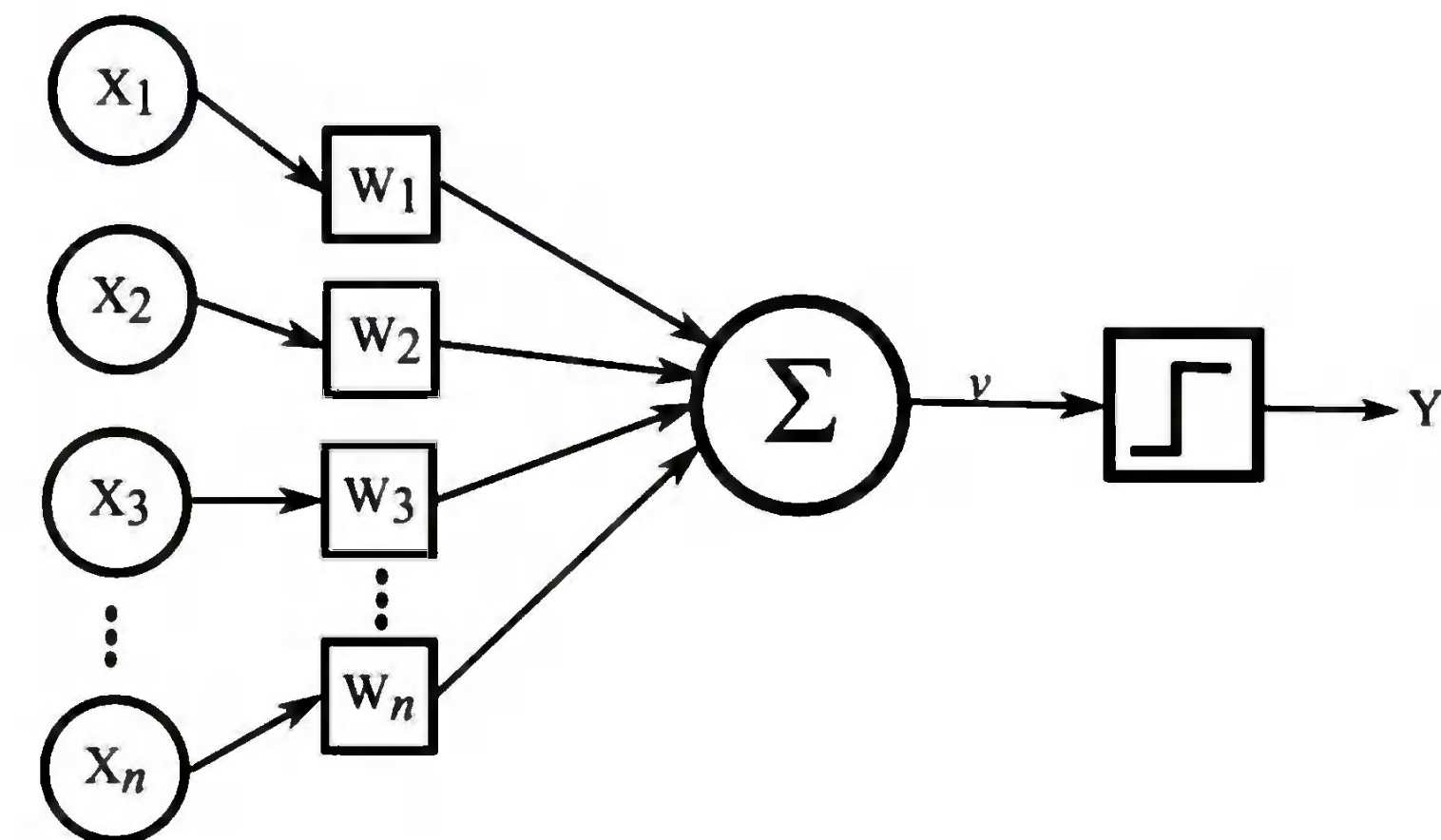


Immagine: il singolo Perceptron di Frank Rosenblatt (1958)

DEEP LEARNING- LIMITI*

MARCUS, G.2018 Deep Learning: A Critical Appraisal

Voracità di dati

A differenza degli esseri umani, che possono imparare astrazioni da pochi esempi, i sistemi di deep learning richiedono grandi quantità di dati di addestramento per apprendere. Ad esempio, gli esseri umani possono comprendere il concetto di "schmister" (sorella di età compresa tra 10 e 21 anni) da una sola definizione, mentre i sistemi di deep learning avrebbero bisogno di molti più esempi.

Superficialità e bassa capacità di trasferimento

Le rappresentazioni apprese dai sistemi di deep learning sono spesso superficiali e non si generalizzano bene a situazioni diverse da quelle di addestramento. Ad esempio, i sistemi di deep learning per i giochi Atari hanno imparato a superare specifici livelli, ma falliscono quando vengono introdotte anche minime variazioni.

Incapacità di rappresentare strutture gerarchiche

Il deep learning fatica a rappresentare e generalizzare strutture gerarchiche complesse, come la sintassi del linguaggio naturale. I sistemi di deep learning basati su reti neurali ricorrenti hanno prestazioni limitate nel rappresentare in modo sistematico la struttura gerarchica delle frasi.

Difficoltà nell'inferenza aperta

I sistemi di deep learning hanno successo in compiti in cui le risposte sono esplicitamente contenute nel testo, ma faticano in compiti che richiedono inferenze basate su conoscenze di background. Ad esempio, i sistemi di lettura automatica hanno successo in compiti di domanda-risposta, ma falliscono in compiti che richiedono l'integrazione di informazioni provenienti da più frasi o la combinazione di conoscenze esplicite e implicite.

Opacità e mancanza di trasparenza

I sistemi di deep learning sono spesso "black box" il cui funzionamento interno non è facilmente interpretabile, rendendo difficile capire come abbiano preso certe decisioni, ad esempio in ambito medico.

Mancanza di integrazione con la conoscenza pregressa

Il deep learning tende a imparare in modo isolato, senza integrare in modo esplicito le conoscenze pregresse. Ad esempio, i sistemi di deep learning per la fisica dei corpi cadenti non incorporano in modo esplicito le leggi di Newton.

Incapacità di distinguere causalità da correlazione

I sistemi di deep learning imparano correlazioni tra input e output, ma faticano a rappresentare esplicitamente le relazioni causali. Un sistema di deep learning potrebbe imparare che altezza e vocabolario sono correlati, senza comprendere che tale correlazione deriva dallo sviluppo cognitivo.

Fragilità e mancanza di robustezza

I sistemi di deep learning sono facilmente "imbrogliabili" e sensibili a piccole perturbazioni dell'input. Ad esempio, sistemi di deep learning per il riconoscimento di immagini possono scambiare facilmente oggetti come auto per frigoriferi se l'immagine viene leggermente modificata.

Difficoltà di ingegnerizzazione

È complesso garantire la robustezza e la replicabilità dei sistemi di deep learning, essendo sistemi difficili da progettare e mantenere. I sistemi di deep learning hanno una maggiore "debito tecnico" (technical debt) rispetto ai sistemi di programmazione tradizionali.

LARGE LANGUAGE MODELS (LLM)

I modelli linguistici di grandi dimensioni (LLM) sono reti neurali avanzate progettate per comprendere e generare linguaggio naturale. Addestrati su ampi set di dati testuali, questi modelli utilizzano tecniche di deep learning per prevedere e produrre testo in modo simile al linguaggio umano, trovando applicazione in numerosi campi, come la traduzione automatica, l'analisi del sentimento e gli assistenti virtuali.

Gli LLM rappresentano il culmine del sogno di un'interfaccia e di una comunicazione con la macchina in linguaggio naturale - un'aspirazione dell'IA sin dai tempi di Alan Turing - Nonostante i progressi, rimangono ancorati a un'elaborazione statistico-sintattica, non raggiungendo ancora una vera comprensione semantica.

Sono numerosi i problemi logici, epistemologici ed implementativi e filosofici che ostacolano la via sulla strada di una comprensione semantica del linguaggio umano (es. il framework problem, la Chinese Room ecc.) come dimostrato dalle varie edizioni del Loebner Prize, una competizione annuale di intelligenza artificiale organizzata dall'Università di Reading in UK che premia il programma di chatbot che meglio imita la conversazione umana, ispirandosi al test di Turing.

Gli output*** degli LLM sono generalizzazioni, in base al data set* e all'addestramento, statistiche randomizzate, che presentano rischi e criticità**

SONO PAPPAGALLI STOCASTICI?

BENDER, E. M., GEBRU, T., McMILLAN-MAJOR, A., SHMITCHELL, S.
2021 On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?,

LINGUAGGIO NATURALE E HCI: POTENZIARE L'INTELLETTTO

Il linguaggio, nella sua dimensione "antropogenica", in relazione all'interfaccia con la macchina, è il concetto chiave per il potenziamento dell'intelletto umano, come anticipato dai pionieri dell'HCI come Douglas Engelbart.

I computer, agendo su un meta-livello linguistico, manipolano simboli (che rappresentano i concetti/relazioni) per strutturare il mondo umano, diventando un medium che trasforma le attività intellettuali rendendole più efficienti.

Il vero potenziale di "augmented intelligence" risiede nello sviluppo simultaneo di computer e utenti, sfruttando il potere del linguaggio naturale per riconfigurare concetti e trasformare il mondo, grazie alla mediazione tecnologica e alla simbolizzazione fornita dal computer come strumento

*Rischi e criticità degli LLMs

Aspetto	Descrizione	Rischi Ambientali	Costi Finanziari
Consumo Energetico	Gli LLM richiedono enormi quantità di acqua ed energia per l'addestramnto e l'esecuzione	Elevate emissioni di CO2 e consumo di acqua dovute al consumo energetico dei data center e ai costi energetici della computazione	Costi elevati per l'infrastruttura e l'energia necessaria per mantenere i server operativi
Hardware e Risorse	Necessità di potenti GPU e altri componenti hardware per il funzionamento.	Impatto ambientale derivante dall'estrazione e dalla produzione di hardware elettronico	Costi significativi per l'acquisto e la manutenzione dell'hardware necessario
Smaltimento dei Rifiuti	Aggiornamenti frequenti dell'hardware generano rifiuti elettronici	Problemi di smaltimento dei rifiuti elettronici che possono contenere materiali pericolosi	Costi di gestione e smaltimento dei rifiuti elettronici.

**Dati e addestramento: rischi e costi

I data set alla base dei LLM sono così vasti e complessi che è difficile per gli umani comprenderne pienamente l'origine, la composizione e il contenuto. Questo può portare a problemi di documentazione, responsabilità e bias.

Aspetto	Descrizione	Rischi	Costi
Cura e Documentazione	Difficoltà nel mantenere una documentazione accurata e trasparente dei dati	Possibile uso di dati inaccurati o non rappresentativi	Costi per sviluppare e implementare sistemi di documentazione e curazione più robusti
Accountability	Difficoltà nel tracciare le fonti dei dati e nell'assicurare la responsabilità per l'uso improprio dei dati	Difficoltà nell'individuare responsabili per errori o danni causati dall'uso dei dati	Costi legali e reputazionali derivanti da potenziali controversie sui dati
Dati Statici e Cambiamenti Sociali	I dati non si aggiornano automaticamente per riflettere i cambiamenti sociali o culturali	Rischio di utilizzare dati obsoleti che non rispecchiano le attuali norme sociali o culturali	Costi associati all'aggiornamento dei set di dati per mantenerli rilevanti e accurati
Bias e Diversità dei Dati	I dati possono contenere bias intrinseci e non garantire una rappresentazione diversificata	Rischi di perpetuare stereotipi e discriminazioni nei modelli addestrati con dati biasati	Costi per identificare e mitigare i bias nei dati

*** Pappagalli Stocastici

Il termine "pappagallo stocastico" si riferisce alla capacità dei modelli LLM di generare testo che sembra significativo senza comprendere effettivamente il contenuto. Questi modelli combinano parole e frasi basandosi su probabilità statistiche, senza un vero processo di ragionamento.

Aspetto	Descrizione	Rischi
Mancanza di Comprensione	I modelli non comprendono veramente il contenuto, limitando la loro capacità di rispondere a domande complesse o ambigue	Rischio di generare risposte inesatte o fuorvianti che potrebbero influenzare decisioni critiche basate su informazioni errate
Bias e Preconcetti	Ripetizione di bias e pregiudizi presenti nei dati di addestramento	Rischio di amplificare bias e stereotipi nei contenuti generati, con potenziali impatti sociali negativi
Dipendenza dalla Probabilità	Generazione di testo basata su probabilità statistiche piuttosto che su un reale processo di ragionamento	Potenziale per output incoerenti o privi di logica che non soddisfano le esigenze dell'utente
Impatto sulla Creatività Umana	Possibilità di influenzare negativamente la creatività e l'originalità umana, affidandosi troppo ai modelli per la generazione di contenuti.	Rischio di una standardizzazione eccessiva dei contenuti, riducendo la diversità e la ricchezza culturale.

KNOWLEDGE CONDITION E COORDINATION NORMS

Gli algoritmi predittivi possono rappresentare una minaccia per la libertà, l'autodeterminazione e la libertà umana, qualora i loro output non siano intesi come generalizzazioni statistiche randomizzate e abbiano le caratteristiche di:

- Bassa precisione:** Le previsioni dell'IA sono più accurate rispetto a quelle umane, ma presentano ancora un alto tasso di errore in molti casi.
- Opacità:** Le persone non comprendono il funzionamento dell'IA, e anche gli esperti trovano gran parte dei suoi processi completamente opachi, coinvolgendo tecniche sconosciute.
- Previsioni estese:** Qualsiasi aspetto della vita di una persona può influenzare le previsioni dell'IA in molte altre aree, come l'assicurazione, l'assunzione lavorativa o l'ammissione scolastica.
- Feedback senza correzione degli errori:** Le decisioni dell'IA influenzano le future valutazioni delle persone, e senza meccanismi di correzione, errori ripetuti creano cicli difficili da interrompere, con i perdenti che rimangono bloccati in ruoli negativi.

Knowledge Condition: Il rispetto della libertà umana richiede di soddisfare, o almeno di avvicinarsi sufficientemente, alla seguente Knowledge Condition: coloro che sono soggetti alle decisioni di altri devono essere in grado, con uno sforzo ragionevole, di sapere che esiste una giustificazione adeguata per tali decisioni e questo deve valere anche per le decisioni agoritmiche

Coordination norms sono regole comportamentali condivise all'interno di un gruppo, basate sulla convinzione comune che la conformità a queste norme sia necessaria per raggiungere un obiettivo comune.

Informational norms sono norme sociali che limitano la raccolta, l'uso e la distribuzione delle informazioni, garantendo che queste pratiche siano trasparenti e giustificate.

Coordination informational norms sono un insieme di regole che regolano l'uso e la gestione delle informazioni in modo etico e responsabile, stabilendo come i dati dovrebbero essere raccolti, utilizzati e distribuiti. Queste norme assicurano la trasparenza nella gestione delle informazioni, permettendo ai consumatori di verificare se le decisioni algoritmiche sono giustificate, soddisfacendo così la Knowledge Condition.

Le coordination norms si creano attraverso una combinazione di educazione e regolamentazione, come campagne educative e regolamenti che promuovono l'uso etico degli algoritmi e stabiliscono standard per le aziende.

Regola	Descrizione
Feedback Correttivo	Stabilisce che i sistemi predittivi devono avere meccanismi adeguati di feedback per correggere eventuali errori e migliorare l'accuratezza delle previsioni
Preparazione dei Dati	Le informazioni nel database devono essere sufficientemente predittive e pertinenti al contesto decisionale, evitando dati irrilevanti o distorsivi
Target Predittivi	I target predittivi devono rispettare standard di equità, spiegabilità, trasparenza e fattibilità per garantire decisioni giuste e comprensibili
Accuratezza dei Dati	Deve esserci una correlazione sufficientemente affidabile tra i dati predittivi utilizzati dal sistema e i risultati prodotti per garantire previsioni precise
Norme Informative di Coordinamento	Le norme devono evolversi con la tecnologia, garantendo trasparenza e adattandosi a nuovi contesti per prevenire usi scorretti o discriminatori degli algoritmi

ARTIFICIAL INTELLIGENCE FOR SOCIAL GOOD (AI4SG)

design/sviluppo/implementazione di sistemi di AI in modo da:

- 1) Prevenire, mitigare o risolvere i problemi che incidono sulla vita umana e/o sul benessere del mondo naturale
- 2) consentire sviluppi preferibili dal punto di vista sociale e/o sostenibili dal punto di vista ambientale

**L'AI4SG APRE ALLA DIMENSIONE
POLITICA DELL'AI**

SUSTAINABLE DEVELOPMENT GOALS



1. Povert  Zero
2. Fame Zero
3. Salute e Benessere
4. Istruzione di Qualit 
5. Parit  di Genere
6. Acqua Pulita e Servizi Igienico-Sanitari
7. Energia Pulita e Accessibile
8. Lavoro Dignitoso e Crescita Economica
9. Industria, Innovazione e Infrastrutture
10. Ridurre le Disuguaglianze
11. Citt  e Comunit  Sostenibili
12. Consumo e Produzione Responsabili
13. Lotta contro il Cambiamento Climatico
14. Vita Sott'Acqua
15. Vita sulla Terra
16. Pace, Giustizia e Istituzioni Forti
17. Partnership per gli Obiettivi

Bias, classificazione, discriminazione

Classificazione

Processo di categorizzazione dei dati in gruppi predefiniti sulla base delle loro caratteristiche. Essenziale per molte applicazioni di MACHINE LEARNING, dove un modello viene addestrato a riconoscere e prevedere pattern e categorie specifiche.



Bias nell'IA e Analisi dei Dati

Pregiudizi o inclinazioni sistematiche che influenzano la raccolta, l'interpretazione o l'elaborazione delle informazioni. Può portare a decisioni distorte non rappresentative della realtà.



Potenziale Distorsione

Scelta delle Categorie:
Le categorie stesse possono riflettere pregiudizi sociali o culturali preesistenti.
Criteri di Classificazione: I criteri di addestramento/dati scelti possono favorire certi gruppi o caratteristiche rispetto ad altri.

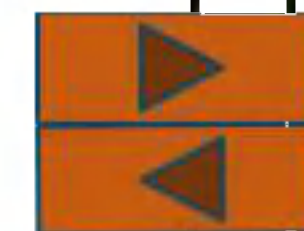


Esempi Storici

Teorie eugenetiche: es. l'idea di migliorare la razza umana attraverso selezioni artificiali. Teorie erano spesso sostenute da dati e ricerche distorte, influenzate da pregiudizi razziali e sociali, portando a politiche discriminatorie e violazioni dei diritti umani.
Test di Intelligenza: Storicamente, i test di intelligenza sono stati criticati per il loro bias culturale, poiché erano spesso progettati per riflettere le conoscenze e le esperienze di gruppi sociali specifici, risultando in valutazioni inique delle capacità intellettuali di persone provenienti da contesti diversi.

Esempi Concreti

Riconoscimento Facciale: Errori più frequenti con pelle scura dovuti a dataset squilibrati.
2. Algoritmi di Assunzione: Bias contro gruppi demografici se addestrati su dati storici distorti.
3. Predictive Policing: Perpetua bias esistenti basandosi su dati storici di pratiche di polizia discriminatorie, riproducendo un ciclo in cui determinate comunità, già sovra-rappresentate nei dati criminali, vengono ulteriormente sorvegliate e criminalizzate
4. Hiring Algorithm: Penalizzava candidature femminili per ruoli tecnici



Discriminazione Razziale

Trattamento ingiusto o svantaggioso basato su razza o origine etnica. In ambito tecnologico, si manifesta attraverso algoritmi che penalizzano determinate etnie a causa di dati distorti o pregiudizi integrati.

Discriminazione di Genere

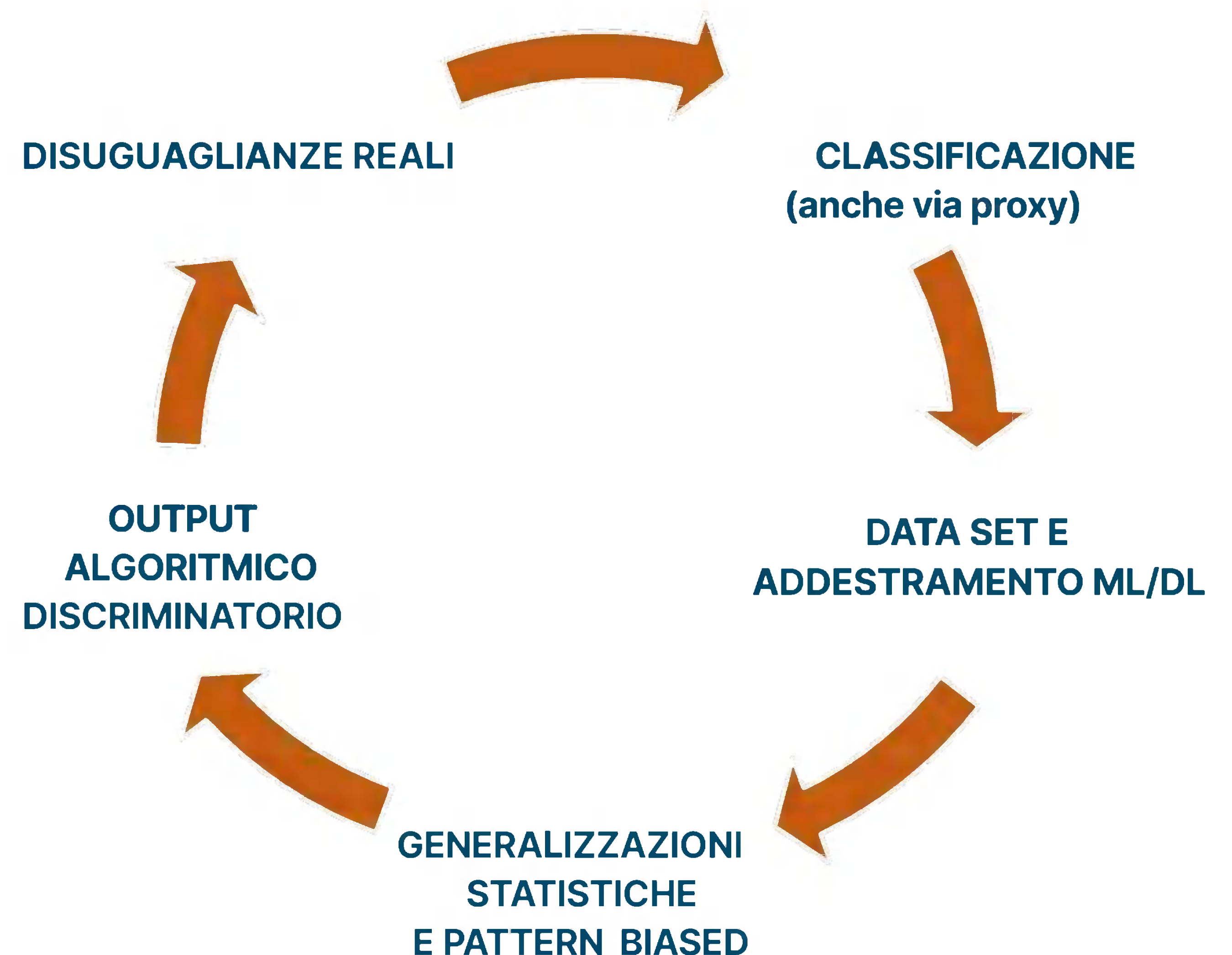
Trattamento iniquo basato sul genere. Nei sistemi di IA, può verificarsi se gli algoritmi sono addestrati su dati che riflettono stereotipi di genere, portando a risultati che perpetuano la disuguaglianza

Loop algoritmico discriminatorio

Le tecnologie algoritmiche non creano e non risolvono da sole le disuguaglianze sociali

Amplificano e riproducono, in forza del loro enorme e crescente potere computazionale e della loro struttura logico-matematica, le iniquità sociali, economiche e politiche reali

Il passo principale per raggiungere l'equità "digitale" è intervenire sul tessuto sociale "analogico"



MAPPING ETICO ALGORITMI

TIPO DI QUESTIONE	CRITICITÀ ETICA	FATTORI DI RISCHIO	ESEMPIO	PRINCIPIO ETICO DI RIFERIMENTO
EPISTEMICA	Prove inconcludenti	Dati o risultati non sufficientemente robusti	AI medica che fa diagnosi basate su correlazioni spurie	Esplicabilità; Non Maleficienza
EPISTEMICA	Prove imperscrutabili	Processi decisionali dell'AI non comprensibili	Algoritmo di trading ad alta frequenza con logica non tracciabile	Esplicabilità; Autonomia
EPISTEMICA	Prove fuorvianti	Generazione di Bias non voluti	Sistemi che riproducono esiti discriminatori, es. giudiziari o economici	Non Maleficienza; Esplicabilità
NORMATIVA	Esiti ingiusti	Decisioni dell'AI che discriminano o penalizzano ingiustamente	Sistema di assunzione che sfavorisce sistematicamente le minoranze	Giustizia; Beneficienza
NORMATIVA	Esiti trasformativi	Cambiamenti sociali profondi causati dall'AI	AI che automatizza lavori su larga scala, causando disoccupazione	Autonomia; Giustizia
NORMATIVA	Tracciabilità	Difficoltà nell'attribuire responsabilità per le azioni dell'AI	Veicolo autonomo coinvolto in un incidente	Esplicabilità; Giustizia

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

1. FALSIFICABILITÀ E IMPLEMENTAZIONE INCREMENTALE

I designer dovrebbero individuare i requisiti falsificabili e testarli in fasi incrementali dall'ambiente di sviluppo al mondo esterno

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; NON MALEFICIENZA

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE*

2. GARANZIE CONTRO LA MANIPOLAZIONE DEI PREDITTORI

I designer dovrebbero adottare garanzie che: 1- assicurino che gli indicatori non causali non distorcano in modo inappropriato gli interventi; 2- limitino, quando appropriato, la conoscenza di come gli input influenzano gli output dei sistemi per prevenirne la manipolazione

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; NON MALEFICIENZA

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

3. INTERVENTO CONTESTUALIZZATO IN FAVORE DEL DESTINATARIO

I designer dovrebbero costruire sistemi decisionali in dialogo con gli utenti che interagiscono con questi sistemi e ne sono influenzati; sulla base della comprensione delle caratteristiche degli utenti, delle modalità di coordinamento, delle finalità e degli effetti di un intervento; e nel rispetto del diritto degli utenti di ignorare o modificare gli interventi.

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; AUTONOMIA

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

4. SPIEGAZIONE CONTESTUALIZZATA IN RAGIONE DEL DESTINATARIO E FINALITÀ TRASPARENTI

I designer dovrebbero scegliere un *livello di astrazione** per la spiegazione del software che soddisfi lo scopo esplicativo auspicato e sia appropriato al sistema e ai destinatari; quindi fornire argomenti che siano razionalmente e adeguatamente persuasivi affinché i destinatari forniscano la spiegazione; e assicurare che l'obiettivo (lo scopo del sistema) per cui viene sviluppato e implementato un sistema di AI4SG sia conoscibile per impostazione predefinita ai destinatari dei suoi risultati.

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; SPIEGABILITÀ

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

4. SPIEGAZIONE CONTESTUALIZZATA IN RAGIONE DEL DESTINATARIO E FINALITÀ TRASPARENTI

I designer dovrebbero scegliere un *livello di astrazione** per la spiegazione del software che soddisfi lo scopo esplicativo auspicato e sia appropriato al sistema e ai destinatari; quindi fornire argomenti che siano razionalmente e adeguatamente persuasivi affinché i destinatari forniscano la spiegazione; e assicurare che l'obiettivo (lo scopo del sistema) per cui viene sviluppato e implementato un sistema di AI4SG sia conoscibile per impostazione predefinita ai destinatari dei suoi risultati.

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; SPIEGABILITÀ

*LIVELLO DI ASTRAZIONE

LIVELLI DI ASTRAZIONE (LoA)

- Un insieme finito ma non vuoto di osservabili (variabili tipizzate interpretate) che sono i blocchi di costruzione di una teoria caratterizzata dalla loro stessa scelta

INTERFACCIA = GRADIENTE DI ASTRAZIONE = UN INSIEME DI LOA IN BASE A REGOLE PRECISE

- NON ACCEDIAMO MAI DIRETTAMENTE AI SISTEMI, MA SOLO MEDIATAMENTE, IN BASE A OSSERVABILI, IN RELAZIONE AI NOSTRI SCOPI, L'ELEMENTO UMANO (E LA SUA SOGGETTIVA FACOLTÀ È ESSENZIALE)
- LA SCELTA DEI LoA È FONDAMENTALE A LIVELLO ETICO PERCHÉ DETERMINA L'IMPEGNO ONTOLOGICO (ES. ANTROPOCENTRICO, BIOCENTRICO, ONTOCENTRICO; RELATIVO AGLI STAKEHOLDER, AI DESIGNER, AI CLIENTI, O AD ALCUNE LORO CARATTERISTICHE/PROPRIETÀ)
- L'INFORMAZIONE È IL LoA ONTOCENTRICO MINIMALE = OGNI NOSTRA INTERAZIONE CON QUALUNQUE ALTRO ENTE (SISTEMA) È MEDIATA DA UNA INTERFACCIA

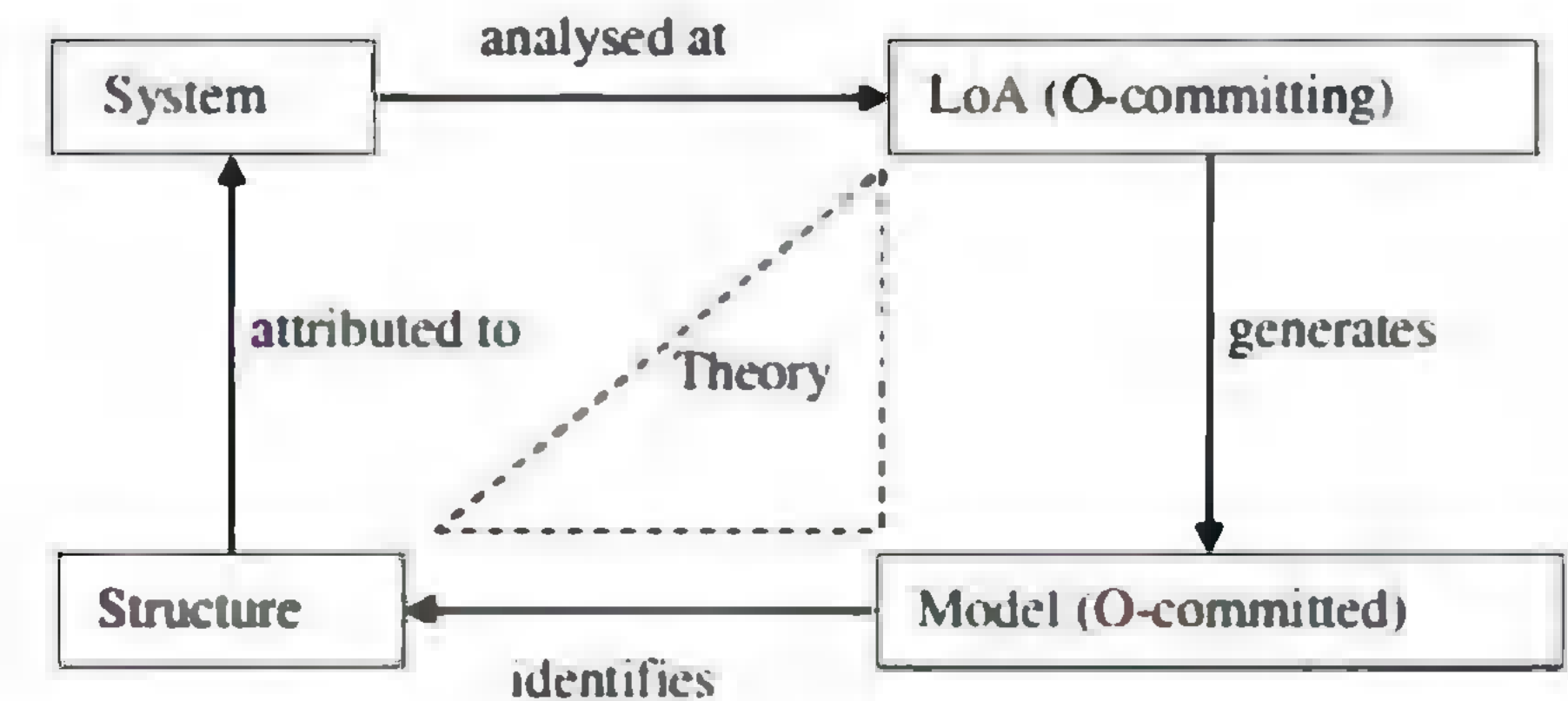
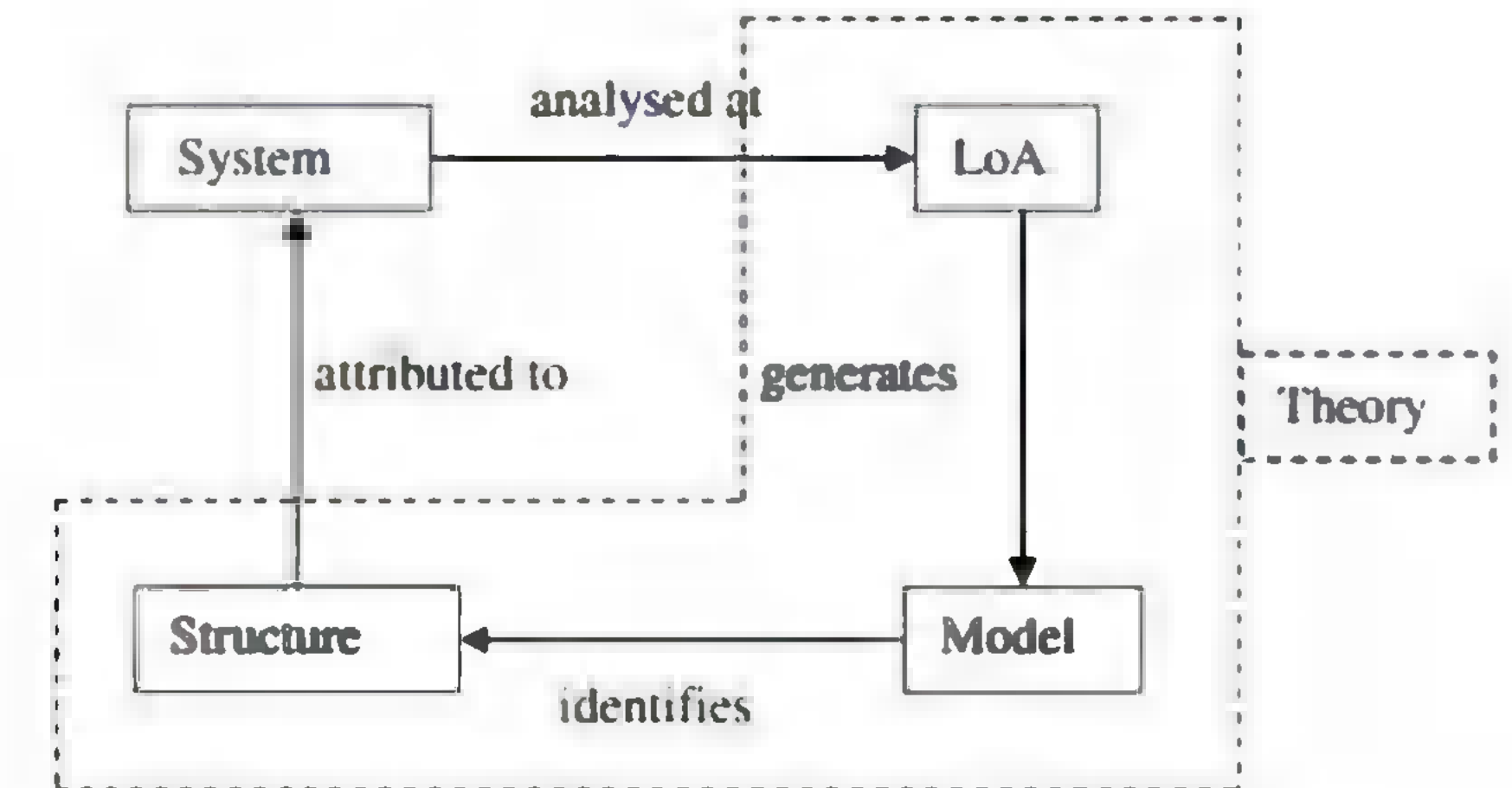


Immagine superiore: Lo schema di una teoria

Immagine inferiore: Lo schema dell'analisi di sistemi a dati LoA considerando l'impegno ontologico

in L. Floridi (2008), *The Method of Levels of Abstraction*, in "Minds and Machines", September 2008, Volume 18, Issue 3, pp 303-329

***LoA E ONTOLOGIA: ESSERE PRESENTI**

L'interazione con tecnologie digitali sempre più potenti rende obsoleto il vecchio concetto di presenza come fallimento epistemico: «essere qui» , grazie a qualche tipo di mediazione tecnologica, in cui la percezione soggettiva non riconosce totalmente il ruolo giocato dalla tecnologia nell'esperienza

**Una presenza “mediata” tecnologicamente ma in cui si perde la percezione del ruolo svolto dalla tecnologia stessa.
(es. immedesimazione, o flow experience, si sarebbe presenti in un libro entro cui “ci si perde”, ma assenti in una teleconferenza)**

**Il modello tradizionale non riesce a dar conto della sempre più pervasiva mediazione tecnologica nell'agire umano
Renderebbe es. piloti via droni, lavoratori in remoto, chirurghi di tele chirurgia ASSENTI = annullerebbe l'accountability per eventuali azioni non etiche**

Dal punto di vista del design questo modello di presenza rende impossibile introdurre quella consapevolezza della mediazione tecnologica (via via sempre più pervasiva, modellata allo scopo e ready-to-hand)e l'introduzione di criteri di feedback imprescindibili per es. la sicurezza

Agenti artificiali non sarebbero presenti malgrado i loro effetti

Nuovo modello: Presenza come osservazione riuscita= sfrutta le potenzialità del digitale per rendere presenti al LoA di enti portatori di proprietà e fonti di cambiamento: un lavoratore da remoto mediante piattaforme smart

*LoA e AGENCY MORALE DEGLI AGENTI ARTIFICIALI

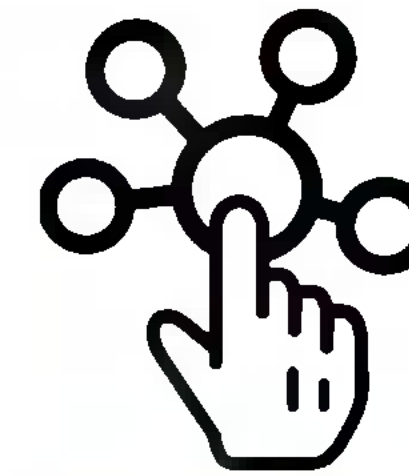
FLORIDI, L., SANDERS, J.

2004 On the Morality of Artificial Agents, Minds and Machines, 14, pp. 349–379.

- LoA minimo per considerare gli **agenti artificiali** (AA) come agenti morali, capaci di produrre conseguenze morali, è costituito da tre proprietà: **Interattività** (1); **Autonomia** (2); **Adattabilità** (3)

MOLTE TECNOLOGIE DI USO COMUNE (es. CHATBOT) AL GIUSTO LoA SONO DA RITENERE PASSIBILI DI PRODURRE CONSEGUENZE MORALI

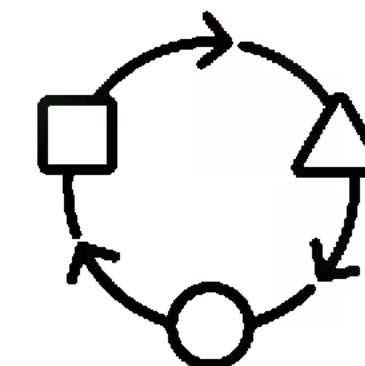
IL CONCETTO STESSO DI AGENCY MORALE (E QUELLO CONNESSO DI ACCOUNTABILITY) NON SONO PIÙ MONOPOLIO ESCLUSIVO DELL' HOMO SAPIENS



1. L'AGENTE E L'AMBIENTE POSSONO AGIRE UNO SULL'ALTRO



2. L'AGENTE PUÒ CAMBIARE STATO SENZA UNA RISPOSTA DIRETTA ALL'INTERAZIONE, MOSTRANDO COMPLESSITÀ E INDIPENDENZA ATTRAVERSO TRANSIZIONI INTERNE CHE RICHIEDONO ALMENO DUE STATI DISTINTI.



3. L'AGENTE PUÒ MODIFICARE LE PROPRIE REGOLE DI TRANSIZIONE IN BASE ALLE ESPERIENZE, APPRENDENDO E ADATTANDOSI NEL TEMPO.

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

5. TUTELA DELLA PRIVACY E CONSENSO DELL'INTERESSATO

I designer dovrebbero rispettare la soglia di consenso stabilita per il trattamento delle raccolte di dati personali

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; AUTONOMIA; NON MALEFICIENZA

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

6. EQUITÀ CONCRETA

I designer dovrebbero rimuovere dagli insiemi di dati rilevanti le variabili e i proxy irrilevanti per un risultato, tranne nel caso in cui la loro introduzione supporti inclusione, sicurezza o altri imperativi etici.

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; GIUSTIZIA

FATTORI E BEST PRACTICES PER PER IL DESIGN E L'IMPLEMENTAZIONE

7. SEMANTIZZAZIONE ADATTA ALL'UMANO

Non ostacolare la capacità delle persone di semantizzare (cioè di dare significato e conferire senso a) qualcosa.

PRINCIPI ETICI DI RIFERIMENTO: BENEFICIENZA; AUTONOMIA

MATRICI ETICHE

MATRICE ETICA è uno strumento analitico utilizzato per valutare le implicazioni etiche delle decisioni e delle azioni in vari contesti, come l'introduzione e l'uso di tecnologie di interattive, digitali e di AI Consente di visualizzare l'impatto su diversi stakeholder e di bilanciare vari principi etici per garantire l'eticità decisioni

- **Step 1: Definire il Contesto**

Identificare la decisione o l'azione che si desidera analizzare. Ad esempio, l'implementazione di un nuovo sistema di AI in un'azienda.

- **Step 2: Identificare gli Stakeholder**

Elencare tutte le parti coinvolte o influenzate dalla decisione. Questi possono includere utenti, clienti, sviluppatori, enti regolatori e la società in generale. Via via si possono approfondire

- **Step 3: Applicare i Principi Etici**

Creare una tabella con gli stakeholder nelle righe e i principi etici nelle colonne.

- **Step 4: Valutare l'Impatto**

Per ogni combinazione di stakeholder e principio, valutare come la decisione influenzerà ciascun stakeholder in termini di benefici, rischi in relazione ai 5 principi etici identificati

- **Step 5: Analizzare e Sintetizzare**

Identificare aree di potenziale conflitto o compromesso e formulare raccomandazioni per affrontare queste sfide.

- **Step 6: Rivedere e Adattare**

Aggiornare la matrice in base ai cambiamenti nelle circostanze o alle nuove informazioni.

Matrice Etica: Principi e Stakeholder: esempio

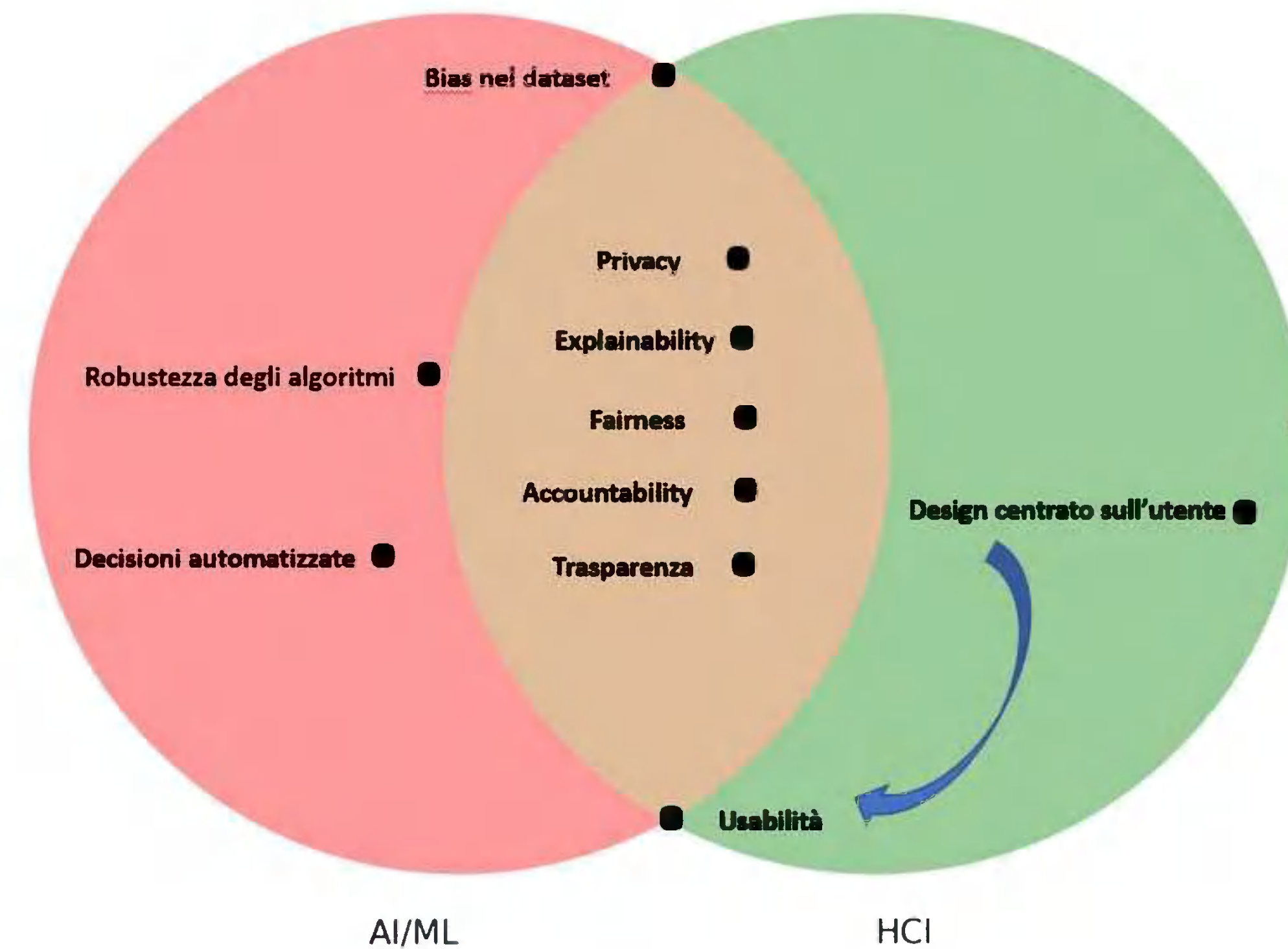
STAKEHOLDER	BENEFICENZA	NON MALEFICENZA	AUTONOMIA	GIUSTIZIA	ESPLICABILITÀ
UTENTI	Miglioramento dell'esperienza utente	Protezione da errori e usi impropri	Controllo e personalizzazione dell'esperienza	Accesso equo ai benefici della tecnologia	Comprensione delle decisioni dell'AI
SVILUPPATORI	Creazione di soluzioni che migliorano la società	Progettazione etica per prevenire danni	Supporto alla capacità decisionale umana	Codifica di algoritmi equi	Documentazione chiara e trasparente
CLIENTI	Aumento della soddisfazione e qualità del servizio	Protezione dei dati personali e sicurezza	Opzioni di personalizzazione e controllo	Accesso equo ai servizi e ai benefici	Chiarezza su come i dati sono usati e protetti
SOCIETÀ	Impatto positivo sull'ambiente e il benessere	Prevenzione di effetti negativi su larga scala	Conservazione dell'autonomia sociale	Lotta alle disuguaglianze ed equità concreta	Trasparenza dei processi
ENTI REGOLATORI	Norme per il benessere pubblico	Regolamentazioni per prevenire abusi	Garanzie per mantenere il controllo umano	Politiche per l'accesso equo alla tecnologia	Richiesta di trasparenza nelle operazioni

Matrice Etica : Principi e Stakeholder: Mock up visualizzazione grafica



Si possono rappresentare le best practices su un grafico cartesiano formato da stakeholder e principi, attribuendo punteggi relative alle importanza delle pratiche etiche, intuitivamente rappresentati dall'area dei riquadri

3. OVERLAPPING ISSUES/FEATURES ETICHE AI(ML)-HCI



- **OVERLAPPING ISSUES/FEATURES ETICHE AI(ML)-HCI**

- **ISSUES/FEATURS ETICHE**

-Privacy

-Privacy by Design

-Tracciabilità: Base per l'attribuzione di responsabilità nei sistemi AI(ML)-HCI

ISSUES/FEATURS ETICHE

FEATURE/ISSUE ETICA	DESCRIZIONE	ESEMPIO
Privacy	Il diritto di mantenere riservati i propri affari personali e le proprie relazioni, limitando l'accesso alle informazioni su se stessi, a meno che non si conceda esplicitamente il permesso ad altri, a garanzia della priopria integrità identitaria*	Il progetto "Living Room of the Future" ha sviluppato DataBox, una piattaforma open-source per sperimentare modelli di elaborazione dei dati personali, permettendo alle persone di gestire i propri dati e l'accesso di terze parti, o il Data Ethics Canvas di Open Data Institute, che aiuta chi implementa e progetta i sistemi a individuare le issues etiche della raccolta dati
Fairness****	Trattamento o comportamento imparziale e giusto, senza favoritismi o discriminazioni, specialmente nel contesto dell'analisi dei dati e del suo impatto, che riduca le caratteristiche generalizzanti potenzialemnte discriminanti della struttura epistemologica degli algoritmi	Sviluppo di modelli di IA che non discriminino determinati gruppi demografici, come assicurare che un'IA per i prestiti bancari non rifiuti le richieste più spesso per le donne rispetto agli uomini. Intefaccia es. sito di prestiti che usa linguaggio chiaro e non richiede altre informazioni (età, sesso, etnia, codice postale ecc.) oltre quelle finanziarie
Accountability	La capacità di un sistema intelligente o automatizzato che utilizza algoritmi di machine learning di spiegare le proprie decisioni in modi che permettano alle persone di credere che siano accurate e corrette.	Intefacce che rendono esplicito il processo decisionale Determinare la responsabilità per gli errori nei veicoli autonomi, come stabilire chi è responsabile quando un'auto a guida autonoma ha un incidente?
Trasparenza	La misura in cui un sistema rende esplicite le proprie decisioni e come queste vengono prese.	L'app Ada Health, dopo aver raccolto i sintomi dell'utente, fornisce possibili condizioni mediche con percentuali di probabilità. Per ogni condizione, l'app mostra quali sintomi inseriti corrispondono, quali non corrispondono e quali informazioni aggiuntive potrebbero essere rilevanti, permettendo all'utente di comprendere il ragionamento dietro la valutazione dell'IA.
Spiegabilità	Progettare sistemi che raccolgono dati e prendono decisioni sulle persone in modo da fornire spiegazioni comprensibili ai non esperti.	Progettare sistemi che raccolgono dati e prendono decisioni sulle persone in modo da fornire spiegazioni comprensibili ai non esperti.

**

PRIVACY*

La privacy si sgancia sempre più dal concetto novecentesco di anonimato/spazio privato
Questa concezione non permette di risolvere questioni di strettissima urgenza e rilevanza come la raccolta di dati i luoghi pubblici (es. videosorveglianza)

L'impatto delle tecnologie digitali ha reso la privacy una questione di **integrità dell'identità personale**. Il soggetto titolare dei dati deve esercitare il consenso sull'uso dei suoi dati personali su cui deve esercitare il controllo

Tramite l'interazione con le tecnologie digitali, che possono aumentare o diminuire la frizione informazionale di cui la privacy è funzione, il soggetto crea la propria stessa identità online e offline

La violazione della privacy non è più paragonabile a una violazione di domicilio, ma a un rapimento digitale

PRIVACY BY DESIGN

Tra le novità e i concetti fondamentali sanciti dal GDPR e dal suo recepimento nella legislazione nazionale, vi è il concetto di **PRIVACY BY DESIGN** sancito dall'art. 25 e dal considerando 78, accanto a quello di **privacy by default**

Privacy by default: i sistemi e le applicazioni devono essere configurati in modo predefinito per garantire il massimo livello di privacy possibile. Questo significa che, senza intervento dell'utente, i dati personali devono essere protetti automaticamente

Impostazioni predefinite	Configurare le applicazioni e i servizi affinché utilizzino impostazioni di privacy più restrittive come standard, richiedendo all'utente di modificare attivamente queste impostazioni per consentire una maggiore condivisione di dati
Limitazione dell'accesso ai dati	Assicurare che solo i dati strettamente necessari per il funzionamento del servizio siano raccolti e che l'accesso sia limitato al personale autorizzato
Eliminazione automatica	Stabilire meccanismi per l'eliminazione automatica dei dati personali non più necessari per le finalità previste

Privacy by design: la protezione dei dati va integrata sin dalle prime fasi di progettazione di sistemi, processi, prodotti o servizi che trattano dati personali

Riduzione al minimo del trattamento dei dati	Limitare la raccolta e il trattamento dei dati personali solo a ciò che è necessario per lo scopo specifico
Pseudonimizzazione	Trasformare i dati personali in modo che non possano essere attribuiti a una persona specifica senza informazioni aggiuntive, riducendo così il rischio di identificazione
Trasparenza	Assicurare che i processi di trattamento dei dati siano chiari e comprensibili per gli interessati, fornendo loro informazioni dettagliate sulle modalità di trattamento
Controllo degli interessati	Consentire agli utenti di gestire e controllare il modo in cui i loro dati vengono trattati, dando loro opzioni per accedere, correggere o eliminare i dati personali
Sicurezza dei dati	Implementare misure di sicurezza appropriate per proteggere i dati personali da accessi non autorizzati, perdite o divulgazioni

****TRACCIABILITÀ: BASE PER L'ATTRIBUZIONE DI RESPONSABILITÀ NEI SISTEMI AI(ML)-HCI**

La tracciabilità degli algoritmi di ML, quando integrata efficacemente nell'HCI, non solo attribuisce responsabilità ma migliora significativamente l'esperienza utente, promuovendo trasparenza, fiducia e un'interazione più informata e consapevole con i sistemi basati su intelligenza artificiale

La tracciabilità facilita la comprensione del funzionamento interno degli algoritmi di ML. Permette di creare interfacce educative che aiutano gli utenti a comprendere meglio come funzionano i sistemi di ML, promuovendo una maggiore alfabetizzazione digitale e una interazione più consapevole con le tecnologie basate su AI.

1. Complessità e Opacità

Aspetto ML: Gli algoritmi di ML possono essere complessi e non trasparenti, rendendo difficile capire come vengono prese le decisioni

Rilevanza HCI: La tracciabilità permette di creare interfacce che spiegano il processo decisionale in modo comprensibile per l'utente, migliorando la trasparenza e la fiducia nell'interazione con il sistema

2. Responsabilità Distribuita

Aspetto ML: Nei sistemi sociotecnici, le decisioni risultano dalle interazioni tra molti agenti, inclusi algoritmi

Rilevanza HCI: La tracciabilità aiuta a progettare interfacce che mostrano chiaramente il ruolo dell'utente e del sistema nel processo decisionale, permettendo una migliore comprensione delle responsabilità condivise

3. Standardizzazione e Regolamentazione

Aspetto ML: La tracciabilità consente di assicurare che gli algoritmi rispettino gli standard etici e legali, fondamentali per aderire a regolamenti come il GDPR

Rilevanza HCI: Permette di sviluppare interfacce che comunicano chiaramente all'utente come i suoi dati vengono utilizzati e protetti, facilitando il consenso informato e la gestione delle preferenze di privacy

4. Risposta e Correzione

Aspetto ML: Identificare e correggere rapidamente gli errori negli algoritmi è possibile con una buona tracciabilità

Rilevanza HCI: Consente di progettare meccanismi di feedback efficaci, permettendo agli utenti di segnalare problemi o risultati inattesi, migliorando così l'interazione e l'accuratezza del sistema nel tempo

5. Etica e Fiducia Pubblica

Aspetto ML: La chiara tracciabilità nel processo decisionale degli algoritmi costruisce fiducia pubblica e rafforza la legittimità delle tecnologie ML

Rilevanza HCI: Permette di creare interfacce che promuovono la trasparenza algoritmica, offrendo agli utenti la possibilità di esplorare e comprendere come le decisioni vengono prese, aumentando così l'accettazione e l'adozione delle tecnologie ML

6. Personalizzazione e Controllo Utente

Aspetto ML: La tracciabilità permette di comprendere come gli algoritmi personalizzano l'esperienza per ciascun utente

Rilevanza HCI: Consente di progettare interfacce che offrono agli utenti un maggiore controllo sulla personalizzazione, permettendo loro di regolare i parametri dell'algoritmo in base alle proprie preferenze e necessità

4. DESIGN HCI

- **LE TRE WAVE DELL'HCI**
- **EVOLUZIONE DELL'HCI: DAL MAINFRAME ALL'UBIQUITÀ...E OLTRE**

-HCI: ontogenesi del soggetto

-Interfacce umane

-Tassonomia delle Interfacce/Esempi di valorizzazione e rischi etici

- **PERSUASIONE E NUDGING**

-Persuasione

-Dark Pattern

-La spinta gentile: verso il pro-ethical design

- **MAPPATURA ETICA DELLE EURISTICHE PER IL DESIGN**

-1 Visibilità dello stato del sistema

-2 Corrispondenza tra sistema e mondo reale

-3 Controllo dell'utente e libertà

-4 Coerenza e standard-*Cognizione-*Esempi di tecniche e consigli di design basati sulle attività cognitive della mente

-5 Prevenzione degli errori- *Modelli mentali

-6 Riconoscimento piuttosto che ricorso

-7 Flessibilità ed efficienza d'uso

-8 Design estetico minimalista

-9 Aiutare gli utenti a riconoscere, diagnosticare e recuperare dagli errori

-10 Aiuto e comunicazione

-Correlazione Principi Etici/Euristiche del Design

LE TRE WAVE DELL'HCI*



anni'80

Si concentra su teorie cognitive e modelli per migliorare le prestazioni degli utenti all'interfaccia.
Utilizza metodi scientifici per testare e falsificare ipotesi

anni'90-2000

Include teorie come l'etnografia, l'antropologia, la cognizione distribuita e la psicologia ecologica. Si espande oltre l'interazione singolo utente-computer per includere contesti sociali e collaborativi.

anni 2000-giorno d'oggi

Si focalizza su valori umani, design, cultura e contesti reali. Esplora come le teorie possono essere applicate per migliorare la vita quotidiana e affrontare sfide globali utilizzando anche AI, interattività, tecnologie smart ecc.

EVOLUZIONE DELL'HCI: dal mainframe all'ubiquità... e oltre

mainframe

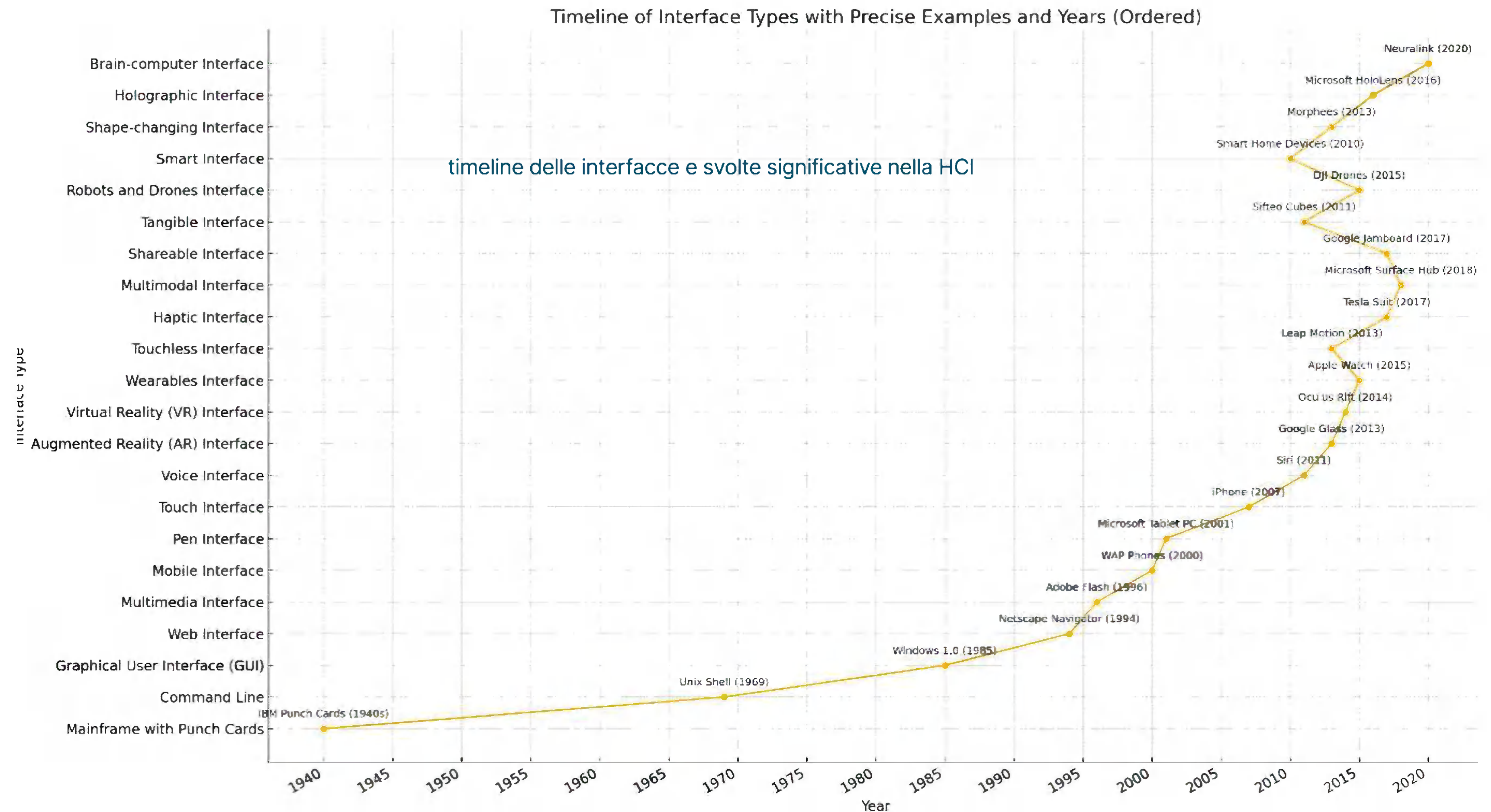
personal computing

mobility computing

ubiquitous computing

...e oltre

L'evoluzione dell'HCI dai primi mainframe ai giorni nostri ha radicalmente trasformato il modo in cui interagiamo con la tecnologia, coinvolgendo sempre più la dimensione corporea. Dai terminali a riga di comando dei mainframe, accessibili solo agli esperti, siamo passati alle interfacce grafiche dei personal computer, che hanno introdotto mouse e tastiere, aprendosi virtualmente a ogni utente. L'era mobile ha portato il computing letteralmente nelle nostre mani, con touchscreen e gesti che hanno reso l'interazione più naturale e intuitiva. Oggi, nell'epoca dell'ubiquitous computing, la tecnologia ci circonda in forme invisibili, dai dispositivi indossabili agli assistenti vocali, coinvolgendo il nostro corpo in modi sempre più sottili e pervasivi. Il futuro promette interfacce cervello-computer e realtà aumentata, prospettando un'integrazione ancora più profonda tra uomo e macchina, dove i confini tra fisico e digitale diventano sempre più sfumati.



HCI: ontogenesi del soggetto

Nella HCI la computazione assume una dimensione ontologica non soltanto a livello esplicativo

I pionieri dell'HCI del "Augmentation Research Center" di Stanford guidati da D. Engelbart negli anni '60 hanno cercato mediante il metodo *bootstrapping* di creare un nuovo tipo di soggetto che interagisce con la macchina: l'utente esperto *augmented*, di cui il cui stesso corpo è posto all'interno dell'interazione con il computer.

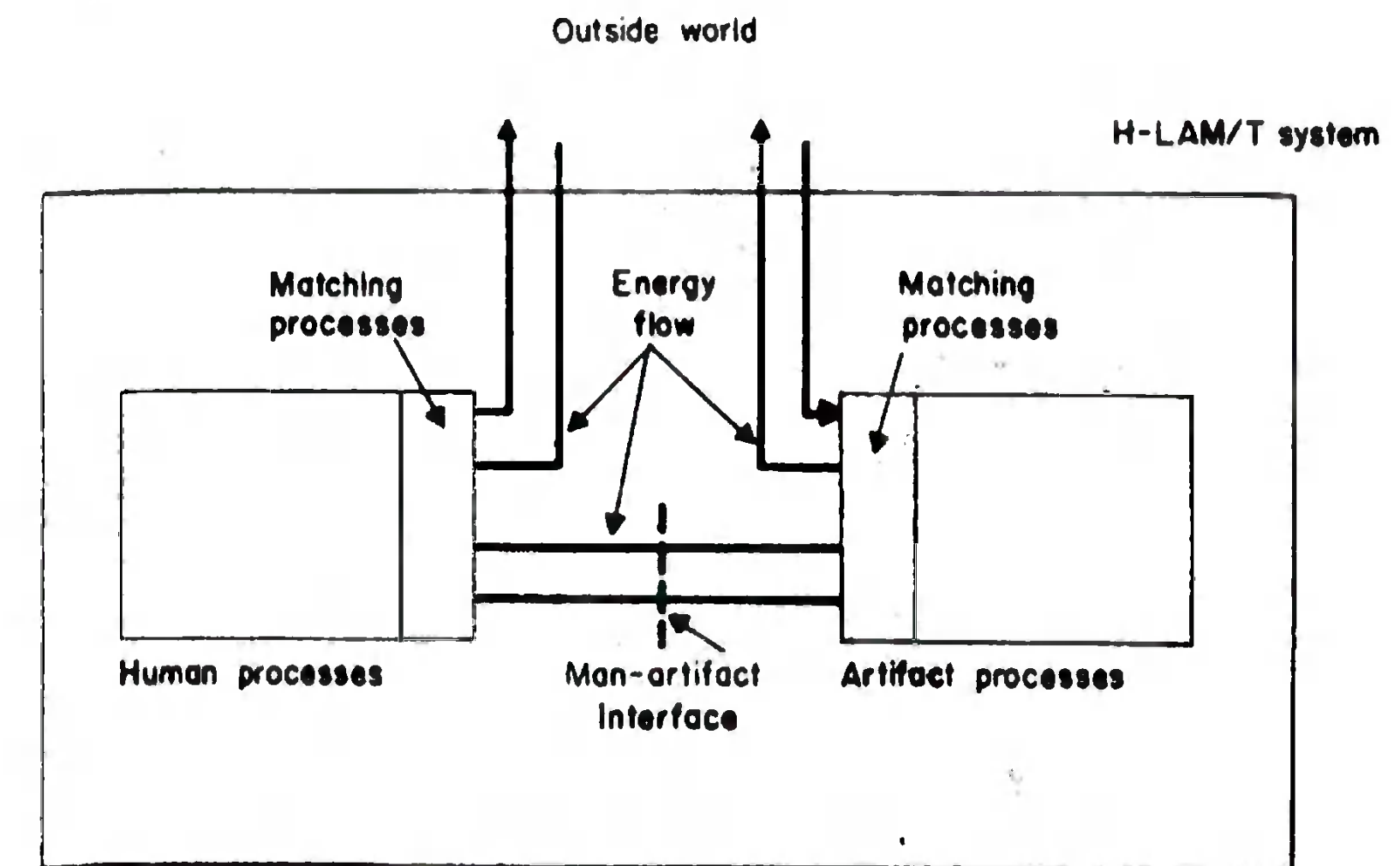
Il rapporto uomo-macchina è anche somestetico

Con le ricerche dello Xerox PARC e la loro diffusione con i computer personali della Apple tra gli anni '70 e '80 il modello di utente cambia: diviene l'utente commerciale generico

il paradigma di interazione diventa la GUI (graphic user interface)

Con l'affermarsi della psicologia nel design di HCI l'utente diviene quello che emerge dai modelli di cognizione, rafforzando l'idea che la costruzione dell'utente spetti al designer che immagina, senza confronto, chi dovrà interagire con i suoi artefatti

Con la critica alla HCI "classica" e l'emergere dell'importanza del contesto socioculturale e valoriale, il design deve tenere in conto le esigenze dell'utente reale personalizzando in modo eticamente orientato la progettazione



L'interfaccia Uomo-Artefatto nel Framework per l'Augmenting Human Intellect del 1962 di D. Engelbart, pioniere del personal computing e inventore del mouse, e lo "Augmentation Research Center" (ARC), situato presso lo Stanford Research Institute (SRI). L'interfaccia uomo computer è vista come un continuo "flusso di energia" in cui l'essere umano e la macchina co-evolvono costruendosi a vicenda

Interfacce umane

Oggi la pervasività delle tecnologie digitali è tale da nascondere la componente di lavoro (e spesso di sfruttamento) di concreti esseri umani.

La narrazione culturale dell'automazione, l'ubiquità di potere computazionale sempre crescente e la promessa di servizi personalizzati sempre a disposizione rischiano di far sparire l'essere umano dietro/dentro la macchina:

Il *wetware* viene occultato dal *software*, il quale nasconde l'*hardware* di silicio e acciaio da cui le tecnologie dipendono. L'essere umano rischia di diventare una "interfaccia umana": il medium che connette *agenti* (artificiali, ibridi) e nostre *risorse* (economiche, dati ecc.)

A partire dalla comune matrice informazionale degli enti che popolano la contemporaneità digitale, il **design** permette, mediante la progettazione **etica** di interfacce, di "estrarre" il **soggetto/utente** dalla macchina (virtuale o reale) valorizzandone, proteggendone e coltivandone **dignità, diritti e prerogative** col fine (già avvertito nei primi studi sulla cibernetica) di un "*uso umano degli esseri umani*"

TASSONOMIA DELLE INTERFACCIE/ESEMPI DI VALORIZZAZIONE E RISCHI ETICI

INTERFACCIA	DEFINIZIONE	ESEMPIO	ES. DI PRINCIPI ETICI VALORIZZATI	ES. DI RISCHI ETICI
Command Line Interface (CLI)	Interfaccia utente che permette l'interazione tramite l'inserimento di comandi testuali	Terminale UNIX, Prompt dei comandi	Autonomia: offre controllo preciso e diretto.	Giustizia: può escludere utenti non tecnici
Graphical User Interface (GUI)	Interfaccia utente che permette l'interazione tramite elementi grafici come finestre, icone e menu	Windows, macOS, GNOME, KDE	Beneficenza: facilita l'uso per un vasto pubblico	Giustizia: può escludere utenti con disabilità visive
Multimedia Interface	Interfaccia che combina testo, immagini, audio e video per fornire un'esperienza interattiva	CD-ROM educativi, Siti Web multimediali	Beneficenza: migliora l'apprendimento e l'engagement; Giustizia: offre canali sensoriali differenziati alle esigenze di ciascuno; Autonomia: ciascun utenete può trovare il canale mediatico a egli più confacente	Esplicabilità: rischio di sovraccarico informativo
Virtual Reality (VR) Interface	Interfaccia che crea un ambiente immersivo tridimensionale con cui l'utente può interagire	Oculus Rift, HTC Vive	Beneficenza: offre nuove possibilità di apprendimento; Giustizia: l'avatar VR può annullare caratteristiche corporee discriminanti	Non maleficenza: rischio di disorientamento
Web Interface	Interfaccia basata su browser che permette l'accesso e l'interazione con contenuti e servizi online	Siti Web, Applicazioni Web	Autonomia: accesso a informazioni e servizi.	Non maleficenza: rischio per la privacy di raccolta dati senza consenso
Mobile Interface	Interfaccia progettata per dispositivi portatili come smartphone e tablet	App per iOS e Android	Autonomia: accesso a servizi ovunque	Autonomia: rischio di dipendenza tecnologica
Appliance Interface	Interfaccia integrata in dispositivi elettronici di uso comune	Smart TV, Forni intelligenti	Beneficenza: semplifica compiti quotidiani	Non maleficenza: rischio per la privacy di registrazione non autorizzata
Voice User Interface (VUI)	Interfaccia che permette l'interazione tramite comandi vocali	Siri, Alexa, Google Assistant	Autonomia: interazione hands-free	Giustizia: può escludere utenti con limitazioni motorie
Pen Interface	Interfaccia che permette l'interazione tramite una penna digitale o uno stilo	Wacom, Apple Pencil	Autonomia: precisione nell'input.	Giustizia: difficoltà per utenti con disabilità motorie
Touch Interface	Interfaccia che consente l'interazione diretta con il display tramite il tocco delle dita o uno stilo	Smartphone, Tablet, ATM	Beneficenza; Autonomia: interazione intuitiva	Giustizia: difficoltà per utenti con disabilità motorie

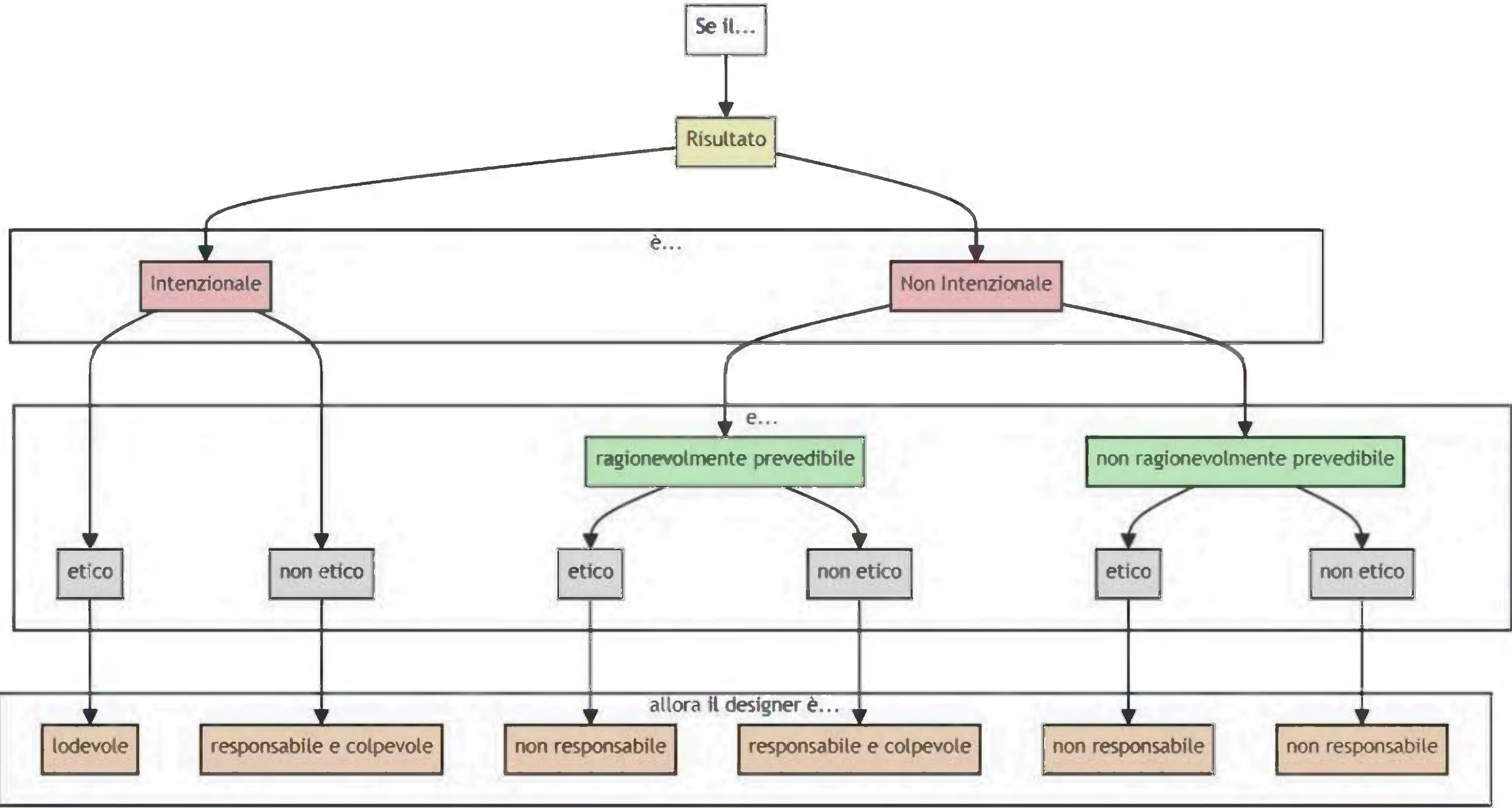
Touchless Interface	Interfaccia che permette l'interazione senza contatto fisico, usando gesti o movimento	Microsoft Kinect, Sensori di movimento	Non maleficenza: riduce il rischio di contaminazione	Non maleficenza: rischio di attivazioni involontarie
Haptic Interface	Interfaccia che fornisce feedback tattile agli utenti tramite vibrazioni o forze	Controller VR, Feedback su smartphone	Non maleficenza: può avvertire di pericoli non visibili	Non maleficenza: possibile sovraccarico sensoriale
Graphical User Interface (GUI)	Interfaccia utente che permette l'interazione tramite elementi grafici come finestre, icone e menu	Windows, macOS, GNOME, KDE	Beneficenza: facilita l'uso per un vasto pubblico	Giustizia: può escludere utenti con disabilità visive
Multimodal Interface	Interfaccia che combina diversi modi di interazione come voce, gesti e tocco	Assistenti vocali con schermo, Smartwatch	Giustizia: aumenta l'accessibilità	Giustizia; Autonomia rischio di escludere utenti con disabilità multiple
Shareable Interface	Interfaccia progettata per essere usata da più persone contemporaneamente	Tavoli interattivi, Schermi collaborativ	Beneficenza: favorisce collaborazione	Autonomia; Giustizia: rischio di esclusione durante l'uso condiviso
Touchable Interface	Interfaccia che permette l'interazione tramite oggetti fisici che l'utente può manipolare	Cubetti programmabili, Bloks	Beneficenza: rende concetti astratti tangibili	Non maleficenza: rischi legati alla manipolazione di oggetti
Augmented Reality (AR) Interface	Interfaccia che sovrappone informazioni digitali al mondo reale attraverso un display	Google Glass, Microsoft HoloLens	Beneficenza: arricchisce l'esperienza del mondo reale.	Non maleficienza: rischio di registrazione non autorizzata; rischi per augmentation non etiche Autonomia: può distorcere l'interazione con la realtà
Wearable Interface	Interfaccia integrata in dispositivi indossabil	Apple Watch, Fitbit	Beneficenza: monitoraggio continuo della salute	Non maleficenza: : raccolta di dati personali sensibili;
Robots and Drones Interface	Interfaccia che permette il controllo e l'interazione con robot o droni.	Droni DJI, Robot domestici	Beneficenza: automazione di compiti complessi; Autonomia; Giustizia: compensare a deficit nell'azione	Non maleficenza: rischio di incidenti fisici; Esplicabilità: difficoltà nell'attribuzione di responsabilità
Brain-Computer Interface	Interfaccia che permette il controllo del sistema tramite segnali cerebrali.	Prototipi di interfacce neurali, es.NeuraLink	Autonomia: può restituire capacità di comunicazione	Non maleficienza; Autonomia. rischio di invasione privacy mentale e distorsione sfera pensiero
Smart Environment Interface	Interfaccia integrata in dispositivi intelligenti che possono automatizzare compiti e apprendere dalle abitudini dell'utente	Smart Home, Termostati intelligenti	Beneficenza: ottimizzazione automatica dell'ambiente	Non maleficenza: raccolta di dati sulle abitudini personali; Autonomia: eccessiva delega all'automazione
Shapechanging Interface	Interfaccia che utilizza ologrammi per creare immagini tridimensionali visibili a occhio nudo	Microsoft Kinect, Sensori di movimento	Beneficenza: adattamento alle esigenze dell'utente	Non maleficenza: rischi legati all'uso di materiali modificabili
Holographic Interface	Interfaccia che utilizza ologrammi per creare immagini tridimensionali visibili a occhio nudo.	Prototipi di ologrammi, HoloLens 2	Beneficenza: visualizzazione 3D senza dispositivi indossabili	Esplicabilità: rischio di rappresentazioni ingannevoli

PERSUASIONE E NUDGING

Berdichevsky, D., & Neuenschwander, E. (1999). Towards an ethics of persuasive technology. Communications of the ACM, 42(5), 51–58.

PERSUASIONE

Il continuo avanzamento tecnologico e il perfezionamento delle tecnologie digitali persuasive, “sistemi informatici interattivi progettati per influenzare atteggiamenti o comportamenti delle persone”, sempre più avanzate e persuasive e l'incorporarsi crescente di valore negli artefatti, rendono indispensabile considerare, nelle nostre valutazioni etiche, anche il design, la sua storia e i progettisti



PRINCIPIO	DESCRIZIONE
Regola d'Oro	Non persuadere gli altri di qualcosa di cui non si acconsentirebbe a essere persuasi.
Principio di Privacy Duale	I creatori devono rispettare la privacy degli utenti almeno quanto la propria.
Principio di Divulgazione	Rivelare motivazioni, metodi e risultati previsti, tranne quando mina un obiettivo etico.
Principio di Responsabilità	I creatori sono responsabili di tutti gli esiti ragionevolmente prevedibili.
Principio di Accuratezza	Le tecnologie non devono disinformare per raggiungere lo scopo persuasivo

DARK PATTERN

Un "dark pattern" nel design è una tecnica utilizzata nelle interfacce utente per ingannare o manipolare gli utenti a compiere azioni indesiderate o non intenzionali

DARK PATTERN	DESCRIZIONE	PRINCIPI VIOLATI	SPIEG. VIOLAZIONE	ESEMPIO
TRICK WORDING	Uso di linguaggio fuorviante o confuso per indurre gli utenti a compiere azioni indesiderate	Non maleficenza, Autonomia, Esplicabilità	Confonde gli utenti, compromettendo la loro capacità di prendere decisioni informate	"Acconsento" invece di "Non acconsento" per la sottoscrizione a una newsletter
SNEAKING	Aggiunta di prodotti al carrello o addebiti nascosti senza il consenso dell'utente	Autonomia, Non maleficenza, Giustizia	Viola la fiducia e l'autonomia dell'utente, causando costi non previsti	Aggiunta di una polizza assicurativa non richiesta durante l'acquisto di un biglietto aereo
OBSTRUCTION	Rendere difficili processi come la disiscrizione per scoraggiare gli utenti	Autonomia, Non maleficenza, Giustizia	Ostacola la libertà dell'utente di interrompere un servizio, spesso causando costi aggiuntivi	Processo lungo e complesso per disiscriversi da un servizio in abbonamento
FORCED ACTION	Forzare gli utenti a compiere azioni per accedere ai servizi (es. registrazione obbligatoria)	Autonomia, Non maleficenza	Costringe l'utente a fornire informazioni o eseguire azioni non desiderate per accedere a un servizio	Richiesta di creare un account per visualizzare contenuti gratuiti
PRIVACY ZUCKERING	Ingannare gli utenti a condividere più dati di quanto desiderino	Autonomia, Non maleficenza, Esplicabilità	Inganna gli utenti facendogli condividere dati personali senza un consenso esplicito e informato	Impostazioni di privacy predefinite che condividono dati con terze parti
BAIT AND SWITCH	Cambiare la funzione attesa di un pulsante o di un'opzione	Autonomia, Non maleficenza, Giustizia	Inganna gli utenti cambiando la funzione di un'azione prevista, causando frustrazione e possibili costi	Un pulsante "Scarica" che in realtà avvia un'installazione di software aggiuntivo

ROACH MOTEL	Rendere facile entrare in un processo ma difficile uscirne (es. iscrizioni)	Autonomia, Non maleficenza, Giustizia	Facilita l'iscrizione ma rende estremamente difficile la disiscrizione, spesso causando costi continui	Sottoscrizione a un servizio che richiede una chiamata telefonica per essere annullata
HIDDEN COSTS	Aggiunta di costi nascosti nel processo di pagamento	Giustizia, Non maleficenza, Autonomia, Esplicabilità	I costi nascosti non vengono rivelati chiaramente, portando a spese inattese per l'utente	Addebiti per spese di spedizione aggiunti solo alla fine del processo di pagamento
CONFIRM SHAMING	Indurre sensi di colpa per dissuadere dall'abbandonare un'azione	Autonomia, Non maleficenza	Manipola le emozioni dell'utente per farlo sentire in colpa se non completa un'azione	Messaggi come "Sei sicuro di voler perdere questa offerta incredibile?" quando si cerca di chiudere una finestra pop-up
DISGUISED ADS	Annunci pubblicitari mascherati da altri contenuti	Esplicabilità, Non maleficenza, Autonomia	Inganna gli utenti facendo passare pubblicità per contenuti genuini, riducendo la trasparenza	Annunci che sembrano post organici sui social media

La spinta gentile: verso il pro-ethical design*

Nudging

Il nudging (“spinta”) è un metodo che altera il comportamento delle persone in modi prevedibili senza proibire alcuna opzione o cambiare significativamente gli incentivi economici

Nudging strutturale/ontologico

Modifica dell'ambiente fisico o dell'architettura delle scelte

Rischi

Può portare a conformità forzata e limitare l'autonomia individuale. Si trasformano le scelte in decisioni predeterminate e limitando la capacità di decisioni critiche

Esempi concreti

1. **Conformità Forzata:** Posizionamento del cibo in mensa: Disporre gli alimenti sani a livello degli occhi nelle mense scolastiche per incoraggiare scelte più salutari. Questo può portare gli studenti a scegliere inconsapevolmente opzioni che non avrebbero altrimenti preferito, limitando la loro libertà di scelta.
2. **Limitazione della Libertà e Autonomia:** Donazione di organi con opt-out. Nei paesi con politiche di donazione di organi con opt-out, i cittadini sono considerati automaticamente donatori a meno che non scelgano di rinunciare esplicitamente. Anche se ciò aumenta i tassi di donazione, potrebbe far sì che le persone partecipino a un sistema di donazione senza aver preso una decisione consapevole o informata.
3. **Potenziata Manipolazione Non Etica:** Posizionare determinati prodotti a livello degli occhi o all'ingresso del negozio può indurre i clienti ad acquistare articoli che altrimenti non avrebbero scelto, influenzando in modo subdolo le loro decisioni di acquisto.

Nudging informativo

Fornire informazioni per influenzare le decisioni

Pregi

Potenzia gli individui supportando l'autonomia e migliorando il processo decisionale informato. Promuove decisioni informate senza interferire direttamente con le scelte individuali

Esempi concreti

1. **Etichette nutrizionali:** Forniscono informazioni chiare sui valori nutrizionali per aiutare i consumatori a fare scelte alimentari consapevoli.
2. **Autovelox:** Informano i conducenti della loro velocità, incoraggiando il rispetto dei limiti senza imporre restrizioni fisiche.

DESIGN → ETHICS BY DESIGN → PRO-ETHICAL DESIGN

*Floridi, L. 2016 Tolerant Paternalism: Pro-ethical Design as a Resolution of the Dilemma of Toleration, SSRN, 1 dicembre 2016. <https://ssrn.com/abstract=3835901>; FRIEDMAN, B., HENDRY, D. G. 2019 Value Sensitive Design: Shaping Technology with Moral Imagination, MIT Press, Cambridge (MA).; THALER, R. H., SUNSTEIN, C. R. 2008 Nudge: Improving decisions about health, wealth, and happiness, Yale University Press, New Haven CT.

MAPPATURA ETICA DELLE EURISTICHE PER IL DESIGN*

1. VISIBILITÀ DELLO STATO DEL SISTEMA

Il sistema deve sempre tenere gli utenti informati su ciò che sta accadendo attraverso un feedback appropriato entro un tempo ragionevole.

es. Fornire un feedback visibile e tempestivo per le azioni degli utenti, come animazioni di caricamento o notifiche di stato come una barra di avanzamento durante il caricamento di una pagina o il download di un file. Creare fiducia mediante una comunicazione continua, anche mediante microinterazioni, fondamentali per dare feedback a tecnologie “a portata di mano”, modellate sul compito (es. per evitare errori di sicurezza)

PRINCIPI ETICI DI RIFERIMENTO:

Esplicabilità: rende chiaro il comportamento del sistema

Autonomia: permette agli utenti di prendere decision informate basate sullo stato del sistema

2. CORRISPONDENZA TRA SISTEMA E MONDO REALE

Il sistema dovrebbe parlare il linguaggio degli utenti, utilizzando parole, frasi e concetti familiari, piuttosto che termini tecnici.

es. Adottare un linguaggio semplice e icone intuitive che riflettano l'esperienza dell'utente nel mondo reale come utilizzare icone riconoscibili e terminologia comune, come una "cartella" per rappresentare una directory.

PRINCIPI ETICI DI RIFERIMENTO:

Explicabilità: facilita la comprensione del sistema utilizzando concetti familiari

Autonomia: consente agli utenti di navigare e utilizzare il sistema in modo intuitivo

3. CONTROLLO DELL'UTENTE E LIBERTÀ

Gli utenti spesso commettono errori e necessitano di un'uscita d'emergenza chiara per lasciare lo stato indesiderato senza dover seguire un lungo processo.

es. Offrire funzioni di annullamento e uscita facili da individuare per consentire agli utenti di correggere gli errori rapidamente come un pulsante "Annulla" o "Indietro" che consente di interrompere un'azione o tornare alla schermata precedente.

PRINCIPI ETICI DI RIFERIMENTO:

Non maleficienza: previene eventuali danni derivanti da errori non intenzionali

Autonomia: dà agli utenti il controllo sulle loro azioni e decisioni

4. COERENZA E STANDARD

Gli utenti non dovrebbero essere costretti a tirare a indovinare se diverse parole, situazioni o azioni significano la stessa cosa. Venir meno alla coerenza può aumentare il carico sulla cognizione* degli utenti.

es. Mantenere un design coerente utilizzando schemi e convenzioni familiari, riducendo l'apprendimento e la confusione come utilizzare un layout di navigazione standardizzato su tutte le pagine di un sito web.

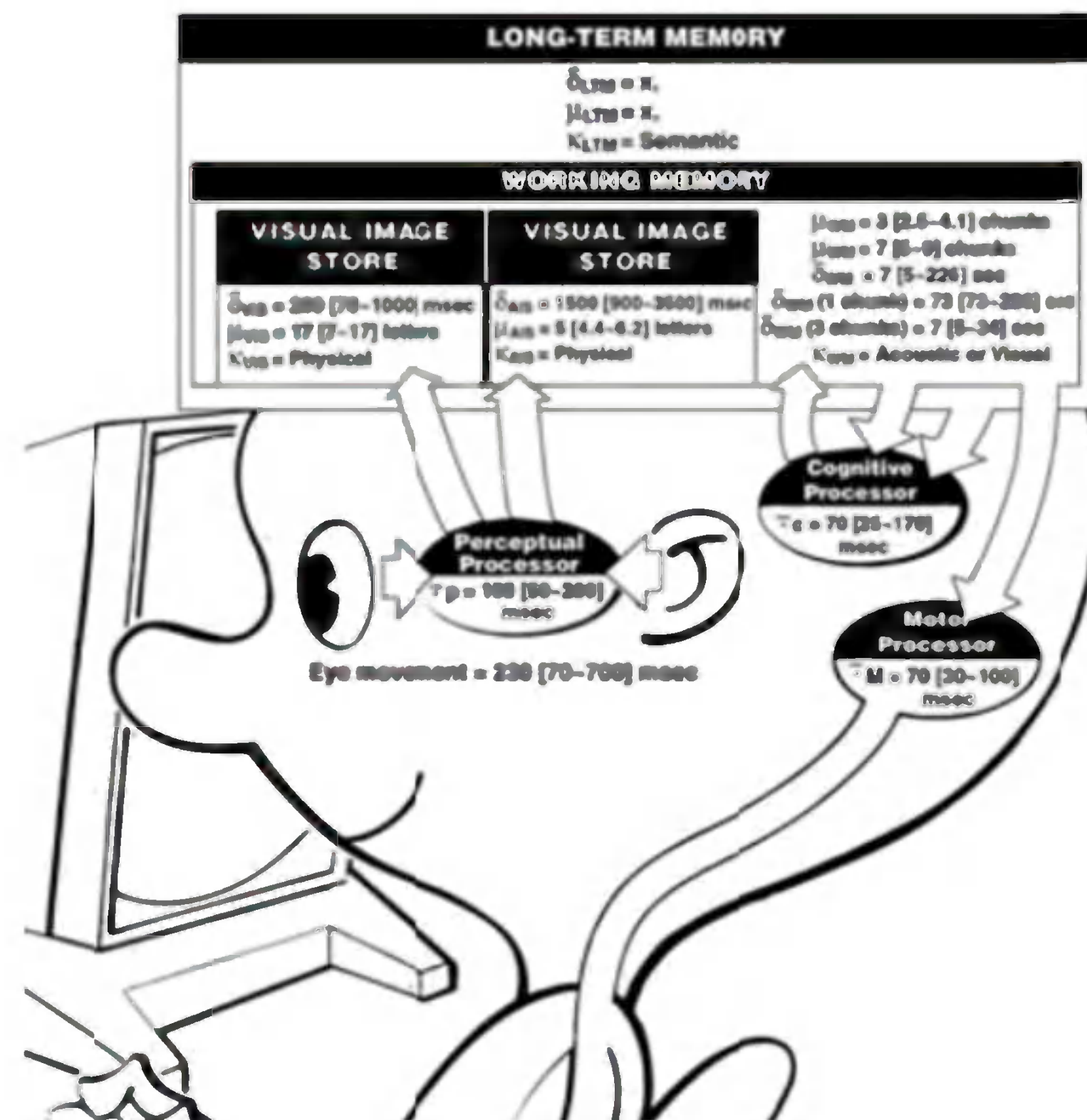
PRINCIPI ETICI DI RIFERIMENTO:

Esplicabilità: rende il sistema più prevedibile e comprensibile

Giustizia: assicura un'esperienza equa per tutti gli utenti, indipendentemente dal loro livello di familiarità con il sistema

COGNIZIONE*

Lo studio sulla cognizione umana ha sempre avuto un ruolo centrale per il design e per l'Interazione Uomo-Macchina, cioè il modellare l'interfaccia sulle strutture cognitive umane, corroborando la visione della **mente come elaboratore di informazioni** conseguenza della nascita delle scienze cognitive, dell'AI e della teoria cibernetica e dell'informazione. L'approccio cognitivo "classico" presenta dei limiti considerando un soggetto sperimentale "neutro", decontestualizzato ed impegnato irrealisticamente in un solo task alla volta



Un esempio storico fondamentale della fase centrata sulla psicologia cognitiva nell'interazione uomo-macchina che veicola una ben determinata idea intorno alla natura della mente e del soggetto è il **modello del processore umano** o MHP (Model Human Processor. Il MHP è un metodo di modellazione cognitiva sviluppato da Stuart K. Card, Thomas P. Moran e Allen Newell nel loro testo seminale *The Psychology of Human-Computer Interaction* del 1983 utilizzato per calcolare quanto tempo viene impiegato ad eseguire un determinato compito.

COGNIZIONE*

Esempi di tecniche e consigli di design basati sulle attività cognitive della mente (**apprendimento, problem solving, pianificazione, decision making, attenzione, percezione e memoria**)

PROCESSI COGNITIVI	LINEE GUIDA	TECNICHE	ESEMPI PRATICI
Attenzione	• Considerare il contesto. Rendere le informazioni salienti quando richiedono attenzione in una determinata fase del compito. • Evitare di sovraccaricare le interfacce visive con troppe informazioni. Ciò si applica specialmente all'uso di colori e grafica. • Supportare efficacemente il passaggio e il ritorno a una particolare interfaccia.	• Utilizzare grafica animata • - Utilizzare colori • - Utilizzare sottolineature • - Ordinare gli elementi • Sequenziare diverse informazioni • Spaziare gli elementi • Utilizzare colori e grafica in modo minimale • Evitare di distrarre o sopraffare l'utente • Utilizzare luci pulsanti • Utilizzare suoni di allerta o voce • Considerare la quantità di informazioni visive e sonore concorrenti	- Evidenziare la notifica di un messaggio non letto con un colore diverso o un'animazione - Utilizzare uno sfondo bianco con testi neri e limitare l'uso di immagini solo a quelle necessarie • Utilizzare un suono di notifica per nuovi messaggi e una luce pulsante per attirare l'attenzione su un'area specifica
Percezione	• Le rappresentazioni delle informazioni devono essere percepite e riconoscibili attraverso diversi media. • Ricercare tecniche di contrasto dei colori. • Utilizzare il feedback aptico con giudizio.	• Progettare icone e rappresentazioni grafiche distinguibili - Utilizzare efficacemente lo spazio bianco - Progettare suoni distinguibili • Scegliere contrasti di colori appropriati - Considerare le deficienze della visione dei colori • Usare tipi di feedback aptico facilmente distinguibili	• Utilizzare icone diverse per 'salva' e 'cancella' in un'applicazione, ben distanziate fra loro • Utilizzare testo giallo su sfondo nero o blu, evitare testo giallo su sfondo bianco o verde • Un feedback di vibrazione breve per un'azione corretta, una vibrazione lunga per un errore

Memoria	• Ridurre il carico cognitivo evitando procedure lunghe e complicate per eseguire i compiti. • Fornire agli utenti vari modi di etichettare le informazioni digitali (ad esempio file, email, immagini) per aiutare a identificare facilmente.	• Promuovere il riconoscimento utilizzando pattern di interazione familiari, menu, icone e oggetti posizionati in modo coerente • Organizzare categorie, colori, etichette, timestamp e icone	• Utilizzare icone standard come il cestino per 'elimina' e la lente d'ingrandimento per 'cerca' • Consentire agli utenti di etichettare le email con categorie colorate o di organizzare i file in cartelle tematiche
Apprendimento	• Progettare interfacce che incoraggino l'esplorazione e tollerino gli errori. • Mantenere i menu e le istruzioni basate sulla voce brevi. • Accentuare l'intonazione delle voci generate artificialmente.	• Guidare gli utenti a selezionare azioni appropriate quando iniziano a imparare • Limitare i menu vocali a tre o quattro opzioni - Evitare istruzioni e direzioni con troppe parti • Usare intonazioni accentuate per voci artificiali • Organizzare categorie, colori, etichette, timestamp e icone	• Offrire suggerimenti interattivi o tutorial iniziali che permettono di provare senza conseguenze • Progettare menu vocali che offrono un massimo di tre scelte per passo • Utilizzare una voce sintetica con intonazioni marcate per differenziare le opzioni in un menu vocale
Risoluzione dei problemi, Pianificazione, Ragionamento e Decision-Making	• Fornire pagine informative facilmente accessibili per chi vuole comprendere meglio come eseguire un'attività in modo più efficace. • Abilitare gli utenti a impostare o salvare i propri criteri o preferenze.	• Utilizzare funzioni semplici e memorabili per supportare decisioni rapide e pianificazione • Consentire la personalizzazione delle impostazioni	• Offrire pagine FAQ facilmente accessibili o guide passo-passo per attività comuni • Permettere agli utenti di salvare le preferenze di ricerca o le impostazioni di visualizzazione •

5. PREVENZIONE DEGLI ERRORI

Meglio progettare il sistema per prevenire gli errori che fornire buoni messaggi di errore. Eliminare condizioni che rendono gli utenti incline a sbagliare o controllarle e presentare agli utenti un'opzione di conferma. Evitare che si commettano sviste (errori inconsci causati da disattenzione) ed errori (sbagli coscienti basati su una discrepanza tra il modello mentale* dell'utente e il design)

es. Implementare controlli di input e opzioni di conferma per ridurre la possibilità di errori, migliorando l'affidabilità del sistema come disabilitare un pulsante "Invia" finché tutti i campi obbligatori non sono compilati correttamente.

PRINCIPI ETICI DI RIFERIMENTO:

EsPLICABILITÀ: migliora la comprensibilità focalizzando l'attenzione sulle informazioni essenziali

Non maleficenza: riduce il rischio di sovraccarico cognitivo e errori dovuti a distrazioni

MODELLO MENTALE*

- **modello mentale:** la rappresentazione mentale che un utente sviluppa di come un sistema funziona, basata sulla loro interazione ed esperienza con esso
- **modello concettuale:** il modello dell'uso dello scopo dell'artefatto a partire da vincoli e affordances
- **modello progettuale:** il modello concettuale del progettista.
- **modello dell'utente:** il modello mentale sviluppato attraverso l'interazione con il sistema.

L'immagine del sistema risulta dalla struttura fisica che è stata costruita (comprese documentazione, istruzioni, etichette). La comunicazione tra designer e user avviene attraverso l'immagine del sistema, immagine e modello che rappresentano ausili per la cognizione fuori dalla mente, la "conoscenza nel mondo".

Se l'immagine del sistema non rende chiaro e coerente il modello progettuale, l'utente finirà per formarsi un modello mentale sbagliato, di qui la difficoltà d'uso e l'errore

6. RICONOSCIMENTO PIUTTOSTO CHE RICORDO

Minimizzare il carico di memoria dell'utente rendendo visibili oggetti, azioni e opzioni. L'utente non dovrebbe dover ricordare informazioni da una parte all'altra del sistema.

es. Progettare interfacce che mostrino opzioni e informazioni rilevanti in modo che gli utenti possano riconoscere piuttosto che ricordare ciò di cui hanno bisogno come mostrare un elenco di pagine visitate di recente per facilitare la navigazione.

PRINCIPI ETICI DI RIFERIMENTO:

Autonomia: facilita l'uso indipendente del sistema

Beneficienza: riduce lo stress cognitivo e migliora l'efficienza dell'utente

7. FLESSIBILITÀ ED EFFICIENZA D'USO

Scorciatoie (invisibili all'utente inesperto) possono spesso velocizzare l'interazione per l'utente esperto in modo tale che il sistema possa soddisfare entrambi.

es. Fornire opzioni di personalizzazione e scorciatoie per utenti esperti, mantenendo al contempo la semplicità per i nuovi utenti come es. combo da tastiera per utenti esperti che velocizzano l'esecuzione di azioni comuni

PRINCIPI ETICI DI RIFERIMENTO:

Giustizia: assicura che il sistema sia accessibile e efficiente per utenti con diversi livelli di esperienza

Beneficienza: migliora la produttività e la soddisfazione degli utenti

8. DESIGN ESTETICO MINIMALISTA

Le interfacce non devono contenere informazioni irrilevanti o raramente necessarie. Ogni unità aggiuntiva di informazione in un dialogo compete con le unità di informazione rilevanti e diminuisce la loro visibilità.

es. Mantenere un design pulito e ordinato, riducendo il sovraccarico di informazioni e mantenendo il focus sugli elementi essenziali.

PRINCIPI ETICI DI RIFERIMENTO:

Esplicabilità: migliora la comprensibilità focalizzando l'attenzione sulle informazioni essenziali

Non maleficienza: riduce il rischio di sovraccarico cognitivo e errori dovuti a distrazioni

9. AIUTARE GLI UTENTI A RICONOSCERE, DIAGNOSTICARE E RECUPERARE DAGLI ERRORI

I messaggi di errore dovrebbero essere espressi in un linguaggio semplice (senza codici), indicare precisamente il problema e suggerire una soluzione.

es. Fornire messaggi di errore chiari e utili, con istruzioni su come risolvere il problema, migliorando l'esperienza complessiva dell'utente come "La password deve contenere almeno 8 caratteri."

PRINCIPI ETICI DI RIFERIMENTO:

Esplicabilità: chiarisce la natura degli errori e come risolverli

Non maleficienza: aiuta gli utenti a superare gli ostacoli e migliorare la loro esperienza

10. AIUTO E COMUNICAZIONE

Anche se è meglio che un sistema possa essere utilizzato senza documentazione, potrebbe essere necessario fornire aiuto e documentazione. Queste informazioni dovrebbero essere facili da cercare, focalizzate sul compito dell'utente, elencare i passi da compiere e non essere troppo lunghe.

es. Creare documentazione di supporto che sia facilmente accessibile e chiara, in modo che gli utenti possano trovare rapidamente le informazioni di cui hanno bisogno come una funzione “cerca” o “aiuto” di facile accessibilità dall'interfaccia principale

PRINCIPI ETICI DI RIFERIMENTO:

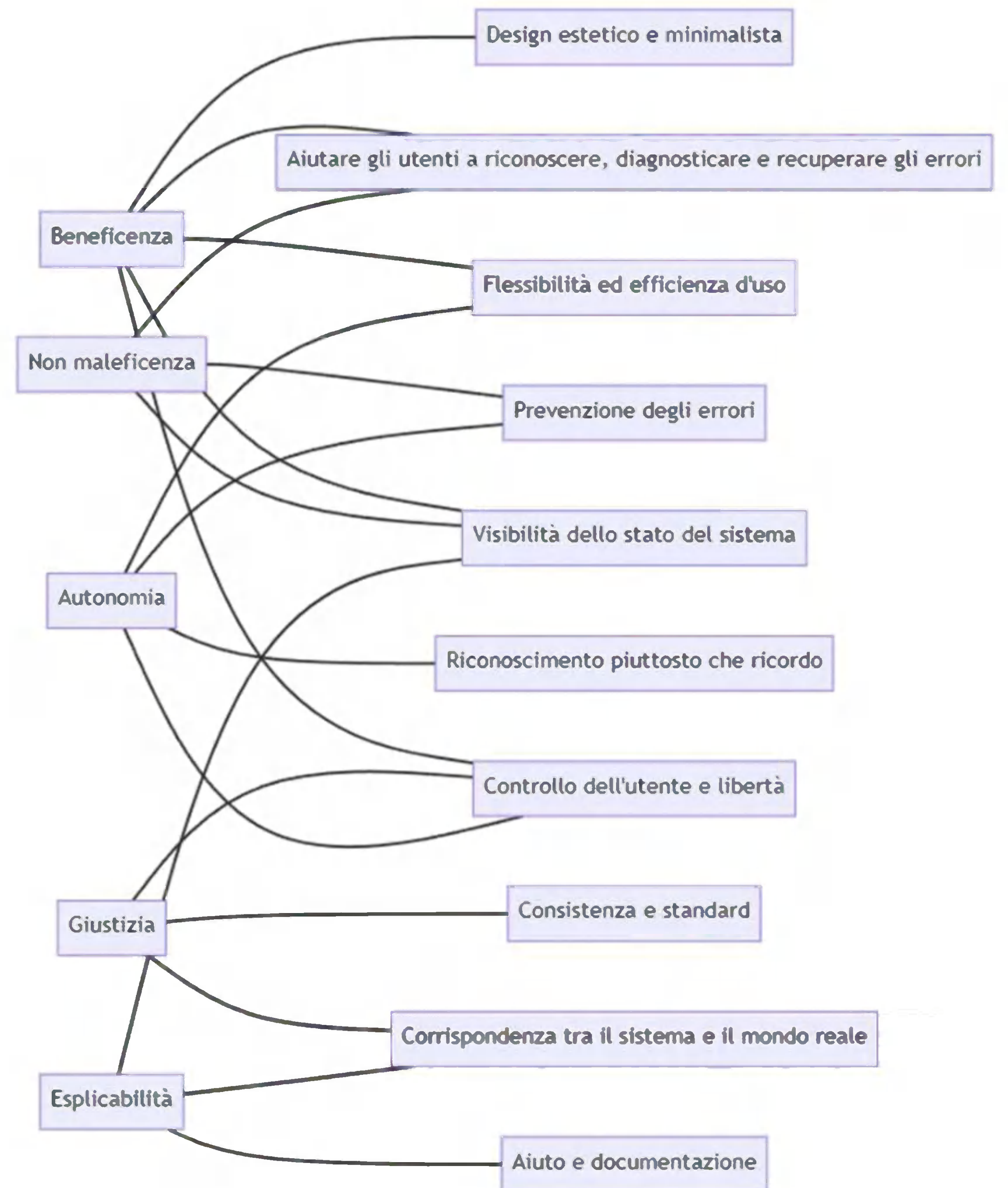
Esplicabilità: fornisce informazioni dettagliate sul funzionamento del sistema

Autonomia: permette agli utenti di apprendere e risolvere i problemi in modo indipendente

CORRELAZIONE PRINCIPI ETICI/EURISTICHE DEL DESIGN

- **Esplicabilità:** Ha una forte correlazione con molte euristiche, riflettendo l'importanza della chiarezza e comprensibilità in quasi tutti gli aspetti del design dell'interfaccia.
- **Autonomia:** Mostra una forte correlazione con le euristiche che promuovono il controllo dell'utente e la libertà di scelta.
- **Non maleficenza:** Ha correlazioni più forti con le euristiche che mirano a prevenire errori o situazioni potenzialmente dannose.
- **Beneficenza:** Mostra correlazioni più forti con le euristiche che migliorano direttamente l'esperienza dell'utente o la sua efficienza.
- **Giustizia:** Ha correlazioni più forti con le euristiche che promuovono l'accessibilità e l'usabilità per tutti gli utenti

IMPLEMENTARE UN DESIGN ETICO, COME DIMOSTRANO LE CORRELAZIONI PRINCIPALI-EURISTICHE (es. riduzione del carico cognitivo)= AUMENTO DELLA PRODUTTIVITÀ



5. DESIGN PER UNA SOCIETÀ EQUA

- **DESIGN PER UNA SOCIETÀ EQUA: AMBIENTE E INCLUSIVITÀ**

Green

- Green Coding
- Linguaggi di programmazione ed efficienza energetica
- Design HCI green: Esempi di best practices per interfacce utente eco-sostenibili

Inclusività: il design per l'equità concreta

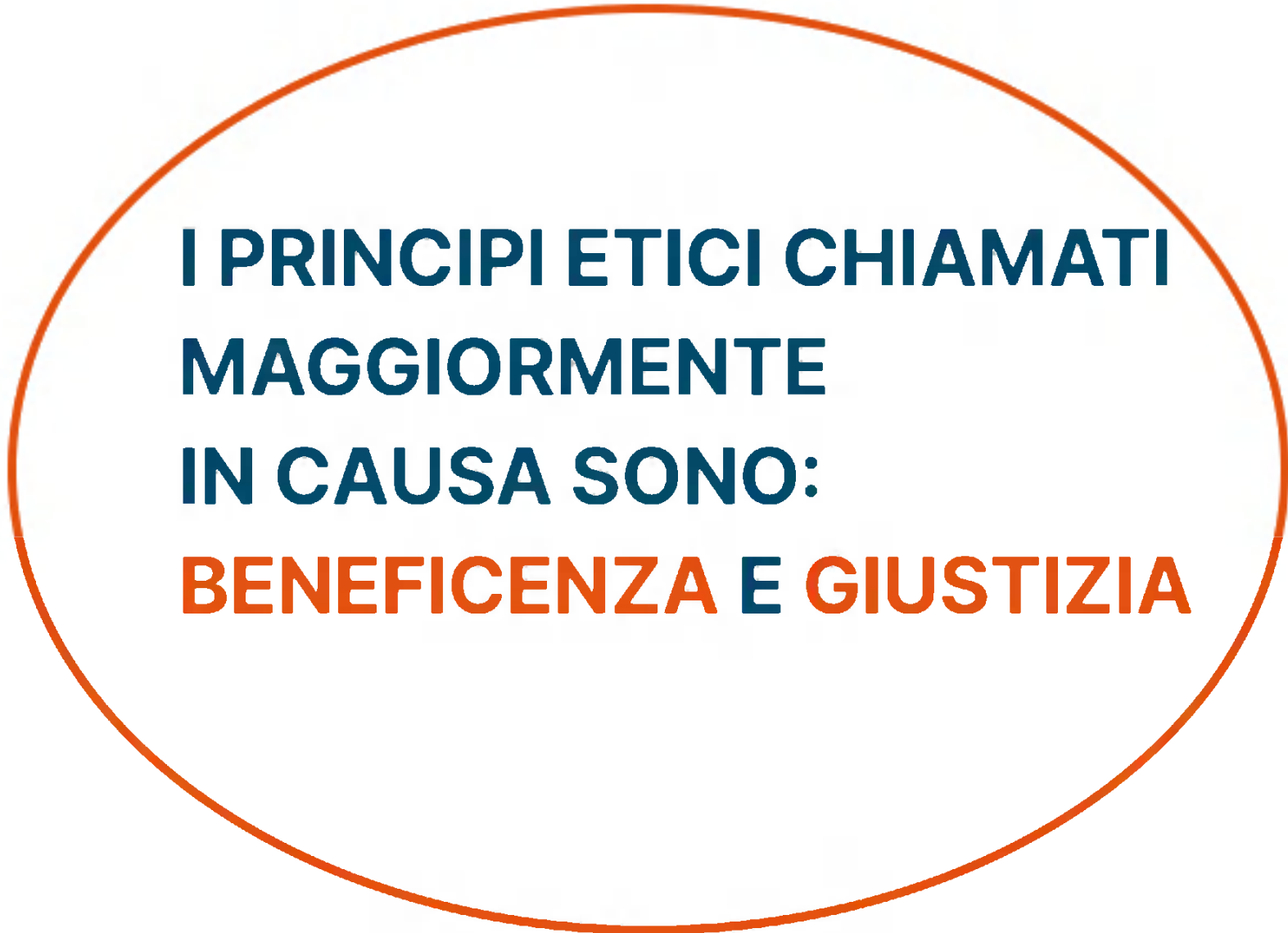
- Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze di genere
- Best Practices ed esempi concreti per il design AI e la Data Science volto a ridurre le disuguaglianze di genere
- Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze basate sull'etnia
- Best Practices ed esempi concreti per il design algoritmico volto a ridurre le disuguaglianze basate sull'etnia

DESIGN PER UNA SOCIETÀ EQUA: AMBIENTE E INCLUSIVITÀ

Le nuove tecnologie dell'Informazione, algoritmiche, digitali e AI rappresentano una nuova forma dell'agire che va progettata eticamente in tutte le sue parti (dall'ideazione, allo sviluppo, all'implementazione e all'interazione con gli utenti) affinché diventi un driver per una società equa e sostenibile al fine di:

- **ridurre gli effetti potenzialmente negativi del suo stesso utilizzo**
- **promuovere gli effetti positivi per ciascuno e per il pianeta**

Due temi particolarmente significativi in cui il **design** dimostra la sua efficacia sono la **riduzione dell'inequità dovuta all'esclusione sociale** di gruppi svantaggiati (gender, minoranze etniche, basso livello socioeconomico, diversamente abili) e il suo uso nella **lotta contro i cambiamenti climatici**



**I PRINCIPI ETICI CHIAMATI
MAGGIORMENTE
IN CAUSA SONO:
BENEFICENZA E GIUSTIZIA**

GREEN

- da un punto di vista epistemico: le tecnologie digitali possono aiutare la nostra comprensione dei fenomeni climatici rendendo possibile la gestione l'elaborazione di quantità enormi di dati e della loro complessità, Questo è possibile con tecnologie AI o con quelle come il Digital Twin che permettono il monitoraggio e la previsione dei fenomeni climatici e delle misure prese a riguardo, come ad esempio Destination Earth, il digital twin del pianeta terra sviluppato dall'EU per la lotta al cambiamento climatico e ora visualizzabile sui device di uso quotidiano
- da un punto di vista operativo: le tecnologie digitali possono offrire soluzioni più ecologiche ed efficaci, per migliorare ed efficientare il consumo di energie e combattere in modo proattivo il climate change



- La prima minaccia ambientale che le tecnologie digitali come l'AI devono affrontare è quella rappresentata dalla loro stessa implementazione e quelle dovute al loro uso
- Si calcola es. che l'addestramento di ChatGpt 3 sia costato nei data center nordamericani (più efficienti dal punto di vista tecnologico di quelli asiatici) 700mila litri di acqua per il raffreddamento dei server
- Una conversazione con una AI LLM ha un costo in termini idrici di circa mezzo litro di acqua
- Studi calcolano che entro il 2027 i data center legati all'intelligenza artificiale consumeranno fino a un massimo previsto di 135 terawattora all'anno, equivalente ai consumi di intere nazioni come l'Argentina o i Paesi Bassi



ACCANTO ALLE SOLUZIONI NORMATIVE SONO NECESSARIE PRATICHE DI DESIGN CHE PONGANO LA SOSTENIBILITÀ NELLO STESSO PROCESSO DI IDEAZIONE, SVILUPPO, IMPLEMENTAZIONE E USO

Green coding: l'approccio alla programmazione che mira a ridurre il consumo energetico e l'impatto ambientale del software. Si concentra sull'ottimizzazione del codice per migliorare l'efficienza energetica, ridurre l'uso delle risorse hardware e minimizzare la carbon footprint associata all'esecuzione di applicazioni software

DESIGN CODING GREEN: ESEMPI DI BEST PRACTICE PER SVILUPPO SOFTWARE SOSTENIBILE

CATEGORIA	BEST PRACTICE	DESCRIZIONE
Ottimizzazione del Codice*	Scrivere codice efficiente	Ridurre il consumo energetico utilizzando algoritmi più veloci e meno complessi per operazioni comuni
Caching dei Dati	Implementare meccanismi di caching	Ridurre l'accesso ripetuto alle risorse. Posizionare il cache vicino all'utente per una migliore efficienza
Uso di CI/CD	Feedback Rapido con CI/CD	Integrare Continuous Integration e Continuous Delivery per velocizzare il feedback e ridurre il consumo energetico
Progettazione Modulare	Suddivisione in moduli	Creare moduli indipendenti che permettano di riutilizzare e ottimizzare solo le parti necessarie, minimizzando gli sprechi
Riduzione dei Tempi di Caricamento	Ottimizzazione dei tempi di caricamento	Monitorare e migliorare i tempi di caricamento e interazione con le applicazioni per ridurre il consumo di energia
Analisi dei Pattern di Utilizzo	Monitoraggio dell'uso delle app	Identificare e ottimizzare le parti delle applicazioni che consumano più risorse, basandosi sul comportamento reale degli utenti
Scelta Tecnologica*	Linguaggi di Programmazione Efficienti *	Scegliere linguaggi che consumano meno energia per specifici compiti.
Risorse e Dipendenze	Ottimizzazione delle Risorse	Ridurre dimensioni di librerie e dipendenze per diminuire il consumo energetico
Architettura del Software	Architettura Attivazione Modulare	Attivare solo componenti necessari per minimizzare sprechi di risorse
Metodologie di Sviluppo	Metodologie di Sviluppo Sostenibili	Adottare pratiche agili e lean per un uso più efficiente delle risorse.
Infrastruttura	Utilizzo del Cloud	Preferire servizi cloud per una maggiore efficienza energetica rispetto ai server tradizionali.

*LINGUAGGI DI PROGRAMMAZIONE ED EFFICIENZA ENERGETICA

La scelta in fase di sviluppo del linguaggio di programmazione può incidere sull'impatto ambientale del software, alcuni linguaggi di programmazione sono più efficienti di altri dal punto di vista del consumo energetico

TOTAL					
ENERGY		TIME		MB	
(c) C	1.00	(c) C	1.00	(c) Pascal	1.00
(c) Rust	1.03	(c) Rust	1.04	(c) Go	1.05
(c) C++	1.34	(c) C++	1.56	(c) C	1.17
(c) Ada	1.70	(c) Ada	1.85	(c) Fortran	1.24
(v) Java	1.98	(v) Java	1.89	(c) C++	1.34
(c) Pascal	2.14	(c) Chapel	2.14	(c) Ada	1.47
(c) Chapel	2.18	(c) Go	2.83	(c) Rust	1.54
(v) Lisp	2.27	(c) Pascal	3.02	(v) Lisp	1.92
(c) Ocaml	2.40	(c) Ocaml	3.09	(c) Haskell	2.45
(c) Fortran	2.52	(v) C#	3.14	(i) PHP	2.57
(c) Swift	2.79	(v) Lisp	3.40	(c) Swift	2.71
(c) Haskell	3.10	(c) Haskell	3.55	(i) Python	2.80
(v) C#	3.14	(c) Swift	4.20	(c) Ocaml	2.82
(c) Go	3.23	(c) Fortran	4.20	(v) C#	2.85
(i) Dart	3.83	(v) F#	6.30	(i) Hack	3.34
(v) F#	4.13	(i) JavaScript	6.52	(v) Racket	3.52
(i) JavaScript	4.45	(i) Dart	6.67	(i) Ruby	3.97
(v) Racket	7.91	(v) Racket	11.27	(c) Chapel	4.00
(i) TypeScript	21.50	(i) Hack	26.99	(v) F#	4.25
(i) Hack	24.02	(i) PHP	27.64	(i) JavaScript	4.59
(i) PHP	29.30	(v) Erlang	36.71	(i) TypeScript	4.69
(v) Erlang	42.23	(i) Jruby	43.44	(v) Java	6.01
(i) Lua	45.98	(i) TypeScript	46.20	(i) Perl	6.62
(i) Jruby	46.54	(i) Ruby	59.34	(i) Lua	6.72
(i) Ruby	69.91	(i) Perl	65.79	(v) Erlang	7.20
(i) Python	75.88	(i) Python	71.90	(i) Dart	8.64
(i) Perl	79.58	(i) Lua	82.91	(i) Jruby	19.84

- Linguaggi di livello più alto hanno una efficienza energetica peggiore rispetto a linguaggi di basso livello di astrazione
- La riduzione del tempo di esecuzione può ridurre il consumo energetico, ma non sempre in modo lineare, poiché il consumo energetico dipende anche dalla potenza assorbita durante l'esecuzione.
- Non c'è una correlazione diretta e consistente tra l'uso di picco della memoria e il consumo energetico della DRAM. È più influente il modo in cui la memoria viene gestita e utilizzata.
- Quando si sceglie un linguaggio di programmazione, è importante bilanciare le esigenze di tempo di esecuzione e uso della memoria con il consumo energetico desiderato.
- Indipendentemente dal linguaggio scelto, l'ottimizzazione del codice per ridurre il tempo di esecuzione e gestire la memoria in modo efficiente può aiutare a migliorare l'efficienza energetica complessiva.
- In applicazioni mobili o dispositivi a batteria, dove la durata della batteria è una priorità, è cruciale scegliere tecniche di programmazione che riducono il consumo energetico senza compromettere significativamente le prestazioni.

La tabella mostra l'efficienza dei linguaggi di programmazione in base ai parametri di Energia, Tempo e Memoria
I valori base = 1 sono i seguenti: l'energia per C è 57,86 J, il tempo per C è 2019,26 ms e Mb per Pascal è 65,96 Mb.



DESIGN HCI GREEN: ESEMPI DI BEST PRACTICE PER INTERFACCE UTENTE ECO-SOSTENIBILI

PRACTICE	DESCRIZIONE
Interfacce Semplici e Chiare	Crea design intuitivi che richiedono meno risorse per essere caricati e utilizzati, migliorando l'esperienza utente e riducendo il consumo energetico
Design Dark Mode	Implementa modalità scure nei design, specialmente per schermi OLED, che consumano meno energia con sfondi scuri
Ottimizzazione delle Immagini	Usa immagini ottimizzate per il web, riducendo risoluzione e formato per diminuire tempo di caricamento ed energia necessaria
Navigazione Efficiente	Progetta percorsi di navigazione che riducono i click e il tempo per trovare informazioni, aumentando l'efficienza
Contenuti Adattivi	Crea contenuti che si adattano al comportamento dell'utente e alle sue preferenze, riducendo l'uso di risorse
Uso Intelligente delle Animazioni	Limita le animazioni a quelle necessarie e usa tecniche che consumano meno energia per gestirle
Efficienza e Minimalismo	Progettare interazioni che richiedono meno elaborazione utilizza un design minimalista per ridurre il consumo di energia
Consapevolezza e Coinvolgimento	Integrare funzioni che informano gli utenti sull'impatto ambientale delle loro azioni, promuovendo abitudini più sostenibili
Pensiero sul Ciclo di Vita	Creare design duraturi che possano essere aggiornati senza bisogno di un rifacimento completo, promuovendo una vita più lunga del prodotto
Riduzione delle Richieste di Rete	Minimizzare le richieste di rete per caricare contenuti, attraverso il pre-caricamento di dati essenziali e la compressione dei dati trasferiti
Accessibilità e Inclusività	Progettare tenendo conto delle esigenze di tutti gli utenti assicura che le applicazioni siano usabili da un pubblico più ampio senza sprechi di risorse

INCLUSIVITÀ: IL DESIGN PER L'EQUITÀ CONCRETA

Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze di genere

BEST PRACTICE	DESCRIZIONE	ESEMPIO
Design inclusivo	Progettare interfacce e prodotti accessibili e utilizzabili da tutti i generi	Apple Watch: offre opzioni di dimensioni e design per adattarsi a diverse preferenze e dimensioni del polso
Rappresentazione equilibrata	Assicurare che immagini, icone e avatar rappresentino equamente tutti i generi	Emoji di professioni: includono versioni maschili e femminili per ogni professione
Linguaggio neutro	Utilizzare un linguaggio che non favorisca un genere rispetto all'altro	LinkedIn: utilizza "loro" invece di "lui/lei" nelle descrizioni dei profili
Test di usabilità diversificati	Includere partecipanti di tutti i generi nei test di usabilità	Google: conduce test di usabilità con gruppi diversificati per garantire che i prodotti funzionino bene per tutti
Personalizzazione dell'interfaccia	Permettere agli utenti di personalizzare l'interfaccia in base alle proprie preferenze	Spotify: permette agli utenti di scegliere opzioni che riflettono la loro identità
Eliminazione degli stereotipi	Evitare stereotipi di genere nella progettazione e nel contenuto	Microsoft Office: ha rimosso il famigerato assistente "Clippy" che era percepito come maschile e fastidioso
Accessibilità	Garantire che le funzionalità siano accessibili a tutti, indipendentemente dalle differenze fisiche	Smartphone: offrono opzioni di accessibilità come l'ingrandimento del testo e il controllo vocale
Feedback diversificato	Raccogliere e implementare feedback da utenti di tutti i generi	Twitter: ha implementato funzionalità anti-molestie basate sul feedback degli utenti, in particolare delle donne
Diversità nei team di sviluppo	Assicurarsi che i team di progettazione e sviluppo siano diversificati	Airbnb: ha implementato politiche per aumentare la diversità di genere nei team tecnici

Includere opzioni non binarie	Non limitare le opzioni di genere a solo "maschio" e "femmina"	Introduzione della terza oprzione “altro”
Utilizzare campi di testo libero	Offrire opzioni che riconoscano identità di genere diverse	Sondaggi online che utilizzavano un campo di testo libero per il genere ha ricevuto risposte positive dalla comunità
Rendere la domanda sul genere opzionale	Non obbligare gli utenti a dichiarare il proprio genere	Permettere agli utenti di saltare la domanda sul genere può ridurre l'ansia e il disagio
Consentire selezioni multiple	Utilizzare caselle di controllo invece di pulsanti radio per le opzioni di genere	Riconoscere che il genere può essere fluido e multiplo, permettendo agli utenti di selezionare più opzioni
Evitare linguaggio escludente	Non usare termini come "altro" o "qualcos'altro" per opzioni non binarie	L'uso di "qualcos'altro" in un sondaggio è stato criticato come de-soggettivante e alienante
Offrire opzioni specifiche e inclusive	Includere opzioni come "donna", "uomo", "non binario", "preferisco non dichiarare" e "preferisco autodescrivermi"	Questa combinazione di opzioni offre un buon equilibrio tra inclusività e praticità per campioni ampi
Considerare il contesto della ricerca	Adattare le domande sul genere in base alla popolazione dello studio e alle dimensioni del campione	Per studi su popolazioni trans, potrebbe essere necessario un approccio in due fasi per identificare partecipanti cisgender e trans
Evitare supposizioni basate sul genere	Non fare ipotesi sulle caratteristiche biologiche dei partecipanti basandosi sul loro genere	Riconoscere che i corpi, come il genere, possono cambiare nel tempo
Aggiornare i sistemi legacy	Rivedere e aggiornare i sistemi esistenti con standard obsoleti	ACM ha recentemente aggiornato i suoi moduli per seguire linee guida più inclusive
Educazione e formazione	Fornire risorse educative e formazione sulla progettazione inclusiva	Google Design: offre corsi online gratuiti sulla progettazione inclusiva
Accessibilità e Inclusività	Non limitare le opzioni di genere a solo "maschio" e "femmina"	CHI 2016 ha ricevuto critiche per offrire inizialmente solo opzioni binarie per i volontari studenti

Best Practices ed esempi concreti per il design AI e la Data Science volto a ridurre le disuguaglianze di genere

FASE DATA SCIENCE	BEST PRACTICE	DEFINIZIONE	ESEMPIO
Business Understanding	Implementare un processo di definizione degli obiettivi che includa prospettive diverse	Assicurarsi che la definizione degli obiettivi di business consideri le prospettive di diversi gruppi demografici	Utilizzare la tecnica del "Reverse Mentoring" dove giovani donne in data science mentorizzano dirigenti senior per comprendere sfide e opportunità uniche
Data Mining	Applicare tecniche di campionamento stratificato per garantire una rappresentazione equa	Utilizzare metodi di campionamento che assicurino una rappresentazione bilanciata di tutti i gruppi demografici nei dati	Utilizzare l'algoritmo SMOTE (Synthetic Minority Over-sampling Technique) per bilanciare dataset con sottorappresentazione di genere
Data Cleaning	Sviluppare e applicare protocolli di de-biasing specifici per il genere	Implementare tecniche per rimuovere o mitigare i bias di genere nei dati prima dell'analisi	Implementare tecniche come il "name masking" per rimuovere identificatori di genere dai dati prima dell'analisi
Data Exploration	Condurre analisi intersezionali considerando il genere insieme ad altre variabili demografiche	Esaminare come il genere interagisce con altre caratteristiche demografiche per influenzare i risultati	Utilizzare la libreria Pandas di Python per creare pivot tables che mostrano metriche chiave segmentate per genere e altre categorie (es. età, etnia)
Feature Engineering	Implementare tecniche di "fairness through unawareness" nella selezione delle feature	Selezionare caratteristiche per i modelli in modo da non favorire ingiustamente alcun gruppo demografico	Utilizzare algoritmi come il "Fair Feature Selection" che ottimizzano sia la performance predittiva che l'equità di genere
Predictive Modeling	Applicare constraint-based learning per garantire equità nei risultati del modello	Utilizzare tecniche di apprendimento automatico che impongono vincoli di equità durante la fase di training	Implementare l'algoritmo "Prejudice Remover Regularizer" che penalizza i modelli che mostrano bias di genere durante il training
Data Visualization	Adottare principi di design inclusivo nelle visualizzazioni dei dati	Creare visualizzazioni che siano accessibili e comprensibili per diverse categorie di utenti	Utilizzare strumenti come Tableau con palette di colori accessibili (es. daltonici) e rappresentazioni non stereotipate dei generi nei grafici

FASE DATA SCIENCE	BEST PRACTICE	DEFINIZIONE	ESEMPIO
Trasversale	Formazione avanzata	Implementare programmi di formazione continua su temi di equità e inclusione nel data science	Implementare un curriculum di "Responsible AI" obbligatorio per tutti i data scientist, con focus su bias di genere e intersezionalità
Trasversale	Recruiting mirato	Sviluppare strategie di reclutamento che promuovano la diversità nel team di data science	Collaborare con organizzazioni come "Women in Data Science" per creare pipeline di talenti diversificate
Trasversale	Mentorship strutturata	Stabilire programmi di mentorship per supportare lo sviluppo professionale di gruppi sottorappresentati	Stabilire un programma di mentorship cross-funzionale con obiettivi KPI misurabili per lo sviluppo di leadership femminile
Trasversale	Valutazione delle prestazioni equa	Implementare sistemi di valutazione che minimizzino i bias inconsci	Implementare revisioni "360 gradi" anonime per ridurre il bias nelle valutazioni delle prestazioni
Trasversale	Cultura di appartenenza	Creare un ambiente di lavoro inclusivo che valorizzi la diversità	Creare "Employee Resource Groups" specifici per donne in data science con budget e supporto esecutivo
Trasversale	Collaborazioni strategiche	Stabilire partnership con istituzioni accademiche e organizzazioni che promuovono la diversità in tech	Partneship con università per sponsorizzare ricerche su metodi di de-biasing in AI/ML
Trasversale	Reporting trasparente	Pubblicare regolarmente metriche sulla diversità e l'inclusione nel team e nei progetti di data science	Pubblicare annualmente un "Diversity in Data Science" report con metriche dettagliate e piani d'azione
Trasversale	Governance etica	Stabilire strutture di governance per garantire l'etica e l'equità nei progetti di data science	Istituire un comitato etico per l'AI con rappresentanza di genere equilibrata per supervisionare tutti i progetti di data science

Best Practices ed esempi concreti per il design HCI volto a ridurre le disuguaglianze basate sull’etnia

BEST PRACTICE	DESCRIZIONE	ESEMPIO
Rappresentazione diversificata	Includere una varietà di etnie nelle interfacce utente, immagini e contenuti	Microsoft Office: inclusione di clipart e immagini stock che rappresentano persone di diverse etnie nei loro modelli di documento
Localizzazione culturale	Adattare contenuti e interfacce a diverse culture e lingue, rispettando le specificità locali	Netflix: personalizzazione delle miniature dei film in base alle preferenze culturali degli utenti in diverse regioni
Riduzione incrementale dei bias e design algoritmico	Progettare sistemi e algoritmi per minimizzare i pregiudizi razziali	Google Photos: revisione e miglioramento degli algoritmi di riconoscimento facciale per ridurre gli errori di identificazione su persone di colore
Diversità nei test di usabilità	Includere partecipanti di diverse etnie nei test di usabilità e ricerche utente	Airbnb: conduzione di test di usabilità con host e ospiti di diverse etnie per identificare e risolvere problemi di discriminazione sulla piattaforma
Linguaggio inclusivo	Utilizzare un linguaggio che sia rispettoso e inclusivo di tutte le etnie	Twitter: aggiornamento della policy linguistica per vietare l'uso di termini razzialmente offensivi
Personalizzazione culturale	Permettere agli utenti di personalizzare l'interfaccia in base alle proprie preferenze culturali	WhatsApp: offerta di pacchetti di sticker che rappresentano diverse culture e festività
Accessibilità linguistica	Fornire contenuti in multiple lingue e dialetti	YouTube: implementazione di sottotitoli automatici in diverse lingue e dialetti
Design per tonalità di pelle diverse	Considerare diverse tonalità di pelle nel design di prodotti e interfacce	Apple- Whatsapp: introduzione di emoji con diverse tonalità di pelle
Feedback e reporting inclusivi	Implementare sistemi di feedback che permettano di segnalare discriminazioni razziali	Facebook: creazione di strumenti specifici per segnalare contenuti che promuovono discriminazione razziale
Educazione anti-razzismo	Incorporare contenuti educativi sull'anti-razzismo nelle piattaforme	LinkedIn Learning: offerta di corsi sul bias inconscio e sulla diversità sul posto di lavoro

Rappresentazione diversificata	Assicurare che i sistemi di intelligenza artificiale funzionino equamente per tutte le etnie	IBM: sospensione produzione sistemi di riconoscimento facciale in quanto intrinsecamente biased verso alcuno gruppi etnici
Rappresentazione equa nei sistemi di IA	Adattare contenuti e interfacce a diverse culture e lingue, rispettando le specificità locali	Netflix: personalizzazione delle thumbnails dei film in base alle preferenze culturali degli utenti in diverse regioni
Design per nomi diversi	Considerare la diversità dei nomi nelle form e nei database	Meta: implementazione di sistemi flessibili per la gestione dei nomi che accolgono diverse convenzioni culturali
Diversità nei team di sviluppo	Promuovere la diversità etnica nei team di progettazione e sviluppo	Google: promessa di implementazione di programmi di reclutamento e retention mirati per aumentare la diversità etnica nei team tecnici
Analisi dell'impatto razziale	Condurre valutazioni dell'impatto dei prodotti su diverse comunità etniche	Conduzione di valutazioni dell'impatto etico dell'IA, includendo considerazioni sull'etnia, prima del rilascio di nuovi prodotti
Collaborazioni con comunità diverse	Coinvolgere diverse comunità razziali nel processo di design	Spotify: collaborazione con artisti e creatori di contenuti di diverse etnie per curare playlist e contenuti culturalmente rilevanti
Supporto per alfabeti non latini	Garantire il corretto supporto per alfabeti e sistemi di scrittura di diverse culture	YouTube, Google Android: miglioramento del supporto per input e visualizzazione di scritture non latine di sottotitoli automatici in diverse lingue e dialetti
Design anti-stereotipo	Evitare la perpetuazione di stereotipi razziali nel design e nei contenuti	Adobe Stock: revisione e aggiornamento del catalogo di immagini per evitare stereotipi razziali
Trasparenza algoritmica	Fornire trasparenza su come gli algoritmi impattano diverse comunità etniche	ProPublica: analisi e pubblicazione dell'impatto razziale degli algoritmi di valutazione del rischio criminale come COMPAS
Governance della diversità	Stabilire strutture di governance per garantire la considerazione della diversità etnica in tutti i progetti	Salesforce: creazione di un "Office of Ethical and Humane Use of Technology" per supervisionare questioni di equità, inclusa l'etnia
Reporting sulla diversità	Pubblicare regolarmente report sulla diversità razziale nell'azienda e nei prodotti	Pubblicare regolarmente report sulla diversità etnica nell'azienda e nei prodotti

Best Practices ed esempi concreti per il design algoritmico volto a ridurre le disuguaglianze basate sull’etnia

BEST PRACTICE	DESCRIZIONE	ESEMPIO
Auditing dei dati per bias etnici	Esaminare sistematicamente i dataset per identificare e mitigare pregiudizi etnici	IBM: Sviluppo di AI Fairness 360, un toolkit open source per rilevare e mitigare bias nei dataset, inclusi quelli etnici
Campionamento stratificato per etnia	Assicurare una rappresentazione equa di tutti i gruppi etnici nei dataset di training	Census Bureau US Gov: Uso di tecniche di campionamento stratificato per garantire una rappresentazione accurata di minoranze etniche nelle indagini nazionali
De-biasing dei dati	Applicare tecniche per rimuovere o ridurre i bias etnici i nei dati prima del training del modello	Google: Implementazione di tecniche di de-biasing nei dataset di immagini per migliorare l'equità nel riconoscimento facciale tra diverse etnie
Fairness constraint in machine learning	Incorporare vincoli di equità etnica negli algoritmi di machine learning	Microsoft: Sviluppo di algoritmi di "fair machine learning" che bilanciano accuratezza e equità etnica nelle previsioni
Test di equità etnica	Valutare sistematicamente i modelli per disparità di performance tra gruppi etnici	ProPublica: Analisi dell'algoritmo COMPAS per valutare disparità etnie nelle previsioni di recidiva criminale
Rappresentazione diversificata nei team	Assicurare diversità etnica nei team di data science e sviluppo algoritmico	Intel: Implementazione di programmi di diversity hiring e retention per aumentare la rappresentanza di minoranze etnie nei team di AI
Documentazione trasparente dei modelli	Documentare chiaramente le considerazioni etniche nella costruzione e valutazione dei modelli	Google Cloud: Introduzione di "Model Cards" che includono informazioni su performance e limitazioni dei modelli per diversi gruppi etnici
Analisi intersezionale	Considerare come l’etnia nteragisce con altre caratteristiche (genere, età, etc.) nell'impatto algoritmico	Pew Research Center: Conduzione di analisi intersezionali su come algoritmi di social media impattano diversamente gruppi etnici e di genere
Monitoraggio continuo per drift etnico	Monitorare costantemente i modelli in produzione per cambiamenti nelle performance tra gruppi etnici	Facebook: Implementazione di sistemi di monitoraggio continuo per rilevare drift nelle performance degli algoritmi di raccomandazione tra gruppi etnici
Inclusione di esperti di equità etnica	Coinvolgere esperti di equità etnica nel processo di sviluppo algoritmico	Airbnb: Collaborazione con esperti di diritti civili per sviluppare algoritmi anti-discriminazione nella piattaforma di prenotazione

Valutazione dell'impatto algoritmico	Condurre valutazioni complete dell'impatto degli algoritmi su diversi gruppi etnici	Governo canadese: Implementazione di Algorithmic Impact Assessments obbligatori per sistemi automatizzati, inclusa la valutazione dell'impatto su gruppi etnici
Tecniche di anonimizzazione razziale	Implementare metodi per nascondere informazioni etniche sensibili nei dati quando non necessarie	NIH US: Sviluppo di tecniche di anonimizzazione differenziale per proteggere l'identità etnica in dataset sanitari
Approcci di "fairness through awareness"	Considerare esplicitamente l'etnia nel design algoritmico per garantire equità	Revisione degli algoritmi di selezione dei candidati per considerare esplicitamente la diversità etnica
Uso di dataset diversificati	Assicurare che i dataset di training includano una rappresentazione equa di tutte le etnie	ImageNet: Aggiornamento del dataset per includere una rappresentazione più equa di immagini di persone di diverse etnie
Metriche di performance specifiche per etnia	Sviluppare e monitorare metriche di performance separate per diversi gruppi etnici	Healthcare.gov: Monitoraggio delle metriche di usabilità e conversione separate per diversi gruppi etnici
Revisione etica pre-deployment	Condurre revisioni etiche con focus sulla razza prima del deployment di algoritmi	DeepMind: Istituzione della divisione Ethics&Society per revisione degli algoritmi, con particolare attenzione all'impatto su gruppi etnici minoritari
Feedback loop inclusivi	Implementare meccanismi di feedback che catturino le esperienze di diversi gruppi etnici	Twitter: Creazione di canali di feedback specifici per comunità etniche sottorappresentate per segnalare problemi algoritmici
Trasparenza algoritmica	Fornire spiegazioni chiare su come gli algoritmi prendono decisioni, specialmente in relazione all'etnia	IDEMIA: Pubblicazione di rapporti dettagliati sulle performance dei loro algoritmi di riconoscimento facciale per diverse etnie
Educazione su bias algoritmici	Formare data scientists, designer e sviluppatori sui bias etnici negli algoritmi	Google: Implementazione di corsi per tutta la comunità di Googlers su temi legati all'equità etnica e contro il razzismo
Collaborazioni con comunità etniche	Coinvolgere diverse comunità razziali nel processo di sviluppo e valutazione algoritmica	Collaborazione con organizzazioni di diritti civili per valutare l'impatto etnico degli algoritmi

6. INCORPORARE L'INNOVAZIONE SOSTENIBILE NELLE COSE: IL DESIGN THINKING E L'ETICISTA DESIGNER

- **DESIGN THINKING**

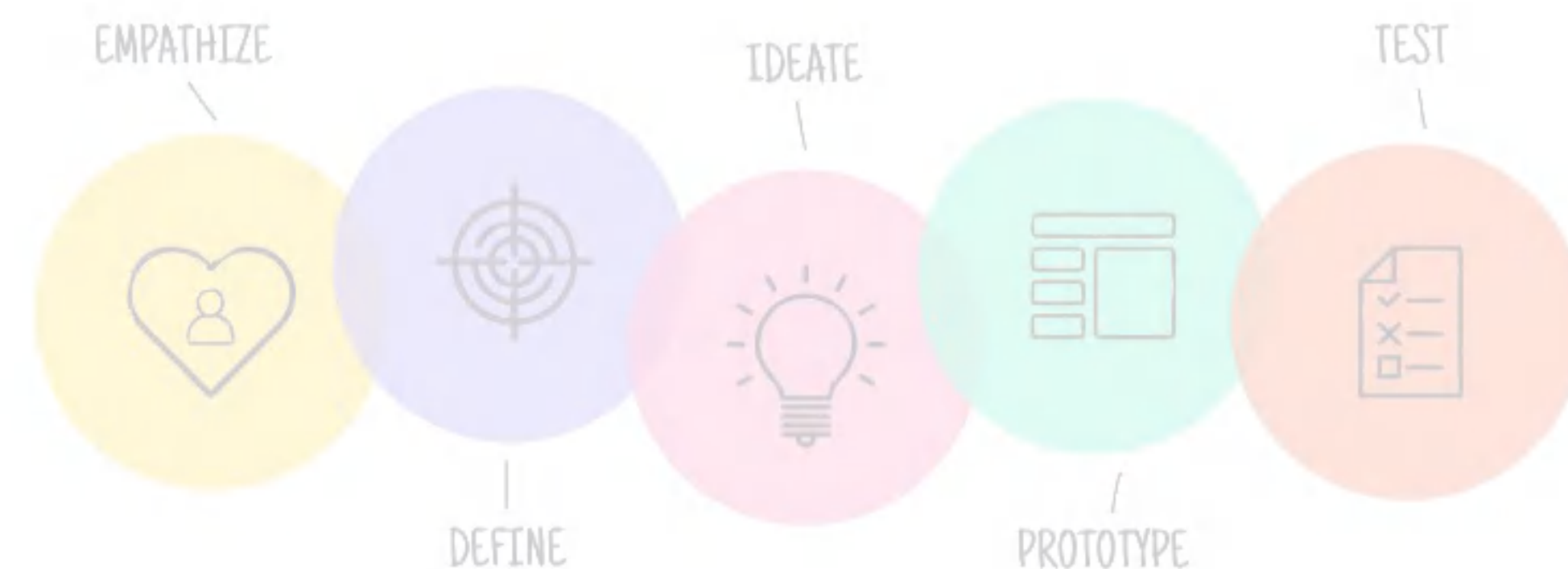
- Discovery
- Esempi di cattive pratiche nella fase di Discovery
- Esempi di buone pratiche nella fase di Discovery - Esempi
- User Personas: rischi e strategie di mitigazione - Esempi
- Prototipation - Esempi
- *Epistemologia Falsificazionista

- **INTRODURRE L'ETICA NEI TEAM DI DESIGN:
L'ETICISTA DESIGNER**

- L'eticista designer
- L'eticista designer nel team: il contributo alla traduzione dei valori in requisiti di progettazione - Esempi

**CONCLUSIONE: FROM TOOL TO.....TOOL
DALLA PAURA ALLA SPERANZA**

INCORPORARE L'INNOVAZIONE SOSTENIBILE NELLE COSE: IL DESIGN THINKING



Discovery	Research		Ideation	Prototipation	Testing
• Comprendere il dominio • Comprendere processi procedure e stakeholder • Individuare gli obiettivi di business	• User personas • Bisogni degli utenti • Win e Pain point	• Cluster Pain & Win Point • Frame the problem	• How Might We • Dot Voting • Priority Map	• Realizzare un prototipo HI-FI dell'idea • Rappresentare l'operatività dell'utente e2e	• Sottoporre il prototipo ad un campione di utenti finali • Osservare, raccogliere feedback e migliorare il prototipo

Design thinking: metodologia basata sul mindset **human-centered** design che ha l'obiettivo ideare e progettare soluzioni a partire da una ricerca sui bisogni degli utilizzatori finali (goal-purpose engineering) in funzione della loro cultura, contesto e capacità cognitiva e motoria con **approccio prototipale a miglioramenti iterativi**

Nel suo essere centrato su bisogni, azioni e caratteristiche di stakeholder reali rispettandone l'autonomia e il concreto contesto e nella sua struttura a feedback di miglioramenti iterativi il **design thinking** è uno strumento privilegiato per la progettazione e l'implementazione di artefatti volti a una una innovazione responsabile e sostenibile. Queste caratteristiche emergono nelle fasi di **discovery** e **prototipation**

Discovery: è il momento iniziale in cui si esplorano e comprendono a fondo le esigenze degli utenti, i problemi da risolvere e il contesto del progetto. Questa fase coinvolge la ricerca, l'osservazione e l'ascolto attivo per raccogliere informazioni e intuizioni utili, ponendo le basi per lo sviluppo di soluzioni innovative e mirate.

esempi di cattive pratiche nella fase di Discovery

FASE	CATTIVA PRATICA	PRINCIPI VIOLATI	ESEMPI
Individuazione degli Stakeholder	Esclusione di gruppi marginalizzati: Se alcuni gruppi di stakeholder, come le persone anziane o con disabilità, vengono ignorati, si rischia di non rispondere alle loro esigenze	Giustizia: Mancanza di equità nel considerare tutti i gruppi coinvolti Non Maleficenza: Escludere gruppi può portare a soluzioni dannose per loro	Un team di design sviluppa es. un'app bancaria senza consultare utenti anziani, portando alla creazione di un'interfaccia difficile da usare per loro
Interviste/Etnografia	Misrepresentazione dovuta a domande mal strutturate: Interviste mal progettate possono distorcere le risposte, portando a interpretazioni errate dei bisogni degli utenti	Beneficenza: Interpretazioni errate possono causare decisioni di design dannose. Accountability: Scarsa responsabilità nella raccolta e analisi dei dati	Un ricercatore fa domande leading durante le interviste, portando a conclusioni fuorvianti su come gli utenti interagiscono con i servizi bancari
Creazione di Journey Map	Stereotipizzazione e semplificazione eccessiva: Rappresentare gli utenti in modo stereotipato o riduttivo può limitare l'efficacia del design. Violazione dell'autonomia: Le journey map che non riflettono accuratamente le attività e preferenze degli utenti compromettono la capacità di questi ultimi di influenzare il design.	Autonomia: Non rispettare le esperienze reali degli utenti limita la loro rappresentazione corretta Trasparenza: Mancanza di chiarezza su come le esperienze degli utenti vengono rappresentate	Creare una journey map per l'app bancaria che presuppone che tutti gli utenti siano esperti di tecnologia, ignorando le difficoltà dei meno esperti, e quindi progetta funzionalità che non sono accessibili a tutti

esempi di buone pratiche nella fase di Discovery

FASE	BUONA PRATICA	DESCRIZIONE	ESEMPI
Individuazione degli Stakeholder	- Inclusività e rappresentanza - Analisi delle reti - Trasparenza	- Assicurarsi che tutti i gruppi di utenti siano rappresentati, incluse le popolazioni marginalizzate o con esigenze speciali - Identificare non solo gli utenti diretti ma anche quelli indiretti	Coinvolgere non solo i clienti tradizionali della banca, ma anche anziani, persone con disabilità e giovani adulti, per comprendere come ciascun gruppo utilizza i servizi bancari digitali
Interviste/Etnografia	- Consenso informato - Neutralità e non-giudizio - Confidenzialità - Rappresentazione accurata - Empatia e rispetto	- Assicurarsi che gli intervistati comprendano come verranno utilizzati i loro dati e rispettare la loro privacy - Mantenere un atteggiamento neutrale e documentare fedelmente le loro risposte	- Prima di intervistare un utente anziano sull'uso dell'app bancaria, spiegare chiaramente l'obiettivo e garantire l'anonimato - Ascoltare senza pregiudizi e documentare le difficoltà incontrate senza interpretarle soggettivamente
Creazione di Journey Map	- Rappresentazione diversificata - Rispetto delle identità - Correttezza - Iterazione e validazione - Consapevolezza dell'impatto sociale	- Assicurarsi che la journey map rifletta le diverse esperienze degli utenti e non solo quelle della maggioranza - Verificare che le map rappresentino accuratamente le esperienze e siano validate in relazione agli utenti	- Creare una journey map che descriva l'esperienza di un utente anziano che utilizza l'app della banca per la prima volta, evidenziando le sfide specifiche. Validare la mappa con feedback diretto dagli utenti anziani e modificarla di conseguenza

User Personas: rischi e strategie di mitigazione

Rischio Etico	Descrizione	Strategie di Mitigazione
Stereotipizzazione	Creazione di personas che riflettono stereotipi, rafforzando idee preconcelte e dannose su certi gruppi di persone	Rappresentare la diversità delle esperienze umane, evitando descrizioni riduttive o stereotipate.
Esclusione	Mancanza di rappresentazione di gruppi diversi, portando alla creazione di soluzioni che non soddisfano le esigenze di tutti gli utenti	Creare personas che riflettano un'ampia gamma di utenti con diverse caratteristiche demografiche e culturali
Semplificazione eccessiva	Riduzione della complessità delle persone reali, portando a decisioni di design che non rispecchiano accuratamente la realtà degli utenti	Bilanciare la necessità di semplificazione con una rappresentazione più contestualizzata
Conferma dei bias	Creazione di personas basate su pregiudizi dei designer, perpetuando stereotipi e disuguaglianze nel design	Basare le personas su ricerche diversificate e dati concreti, rivedendo costantemente i bias personali
Etichettamento	Attribuzione di etichette rigide a gruppi di utenti, trascurando le differenze individuali e promuovendo soluzioni standardizzate	Riconoscere la fluidità delle identità e aggiornare le personas con nuove informazioni e prospettive
Impatto sociale	Influenza negativa sulla percezione sociale di certi gruppi, rafforzando rappresentazioni riduttive o negative nella cultura più ampia	Sensibilizzare sul potenziale impatto sociale del design e adottare un approccio inclusivo e rispettoso

Prototipation: consiste nel creare versioni semplificate e a basso costo del prodotto o servizio per testare e validare le soluzioni proposte. Questo processo permette di esplorare idee, individuare eventuali problemi e raccogliere feedback utili dagli utenti, migliorando progressivamente il progetto prima della realizzazione finale.

La prototipazione è essenziale per affrontare le incertezze legate alle nuove tecnologie introdotte nella società. Epistemologicamente*, si basa sull'identificazione di requisiti falsificabili, che vengono testati e migliorati iterativamente per affinare l'artefatto. Questo processo non solo verifica il funzionamento tecnico, ma valuta anche l'impatto sociale e comportamentale, assicurando che le soluzioni rispettino valori etici come beneficenza, non maleficenza, autonomia, giustizia e trasparenza, riducendo i rischi prima del lancio su larga scala, riducendo in modo graduale e centrato sui feedback degli stakeholder le tensioni nei vari contesti che possono nascere con l'introduzione di un nuovo artefatto.

La prototipaton è una delle chiavi di volta per una innovazione tecnologica sostenibile e responsabile

- **Beneficenza:** La prototipazione consente di creare soluzioni che migliorano il benessere degli utenti, identificando e risolvendo problemi reali.
- **Non maleficenza:** Aiuta a minimizzare i rischi, permettendo di testare e correggere potenziali danni prima del lancio.
- **Autonomia:** Rispetta l'opinione degli utenti coinvolgendoli nel processo di sviluppo e di miglioramento dei mock-up in base ai loro feedback
- **Giustizia:** Garantisce che le soluzioni siano eque e inclusive per tutti gli utenti coinvolti, valutando l'impatto graduale dei vari prototipi dell'artefatto
- **Trasparenza e Accountability:** Documenta chiaramente il processo, assicurando responsabilità e chiarezza nelle decisioni progettuali e rispetta dei criteri epistemologici chiari

*Epistemologia falsificazionista

Il processo di messa alla prova e miglioramento iterativo che struttura la fase della **prototipation**, si basa sulla epistemologia falsificazionista, che procede per **trial and error**, elaborata da Karl Popper.

Prendendo le mosse dai limiti del verificazionismo e dei procedimenti induttivi, per il paradigma falsificazionista una teoria non si verifica (es. con un esperimento), ma viene testata al fine di essere falsificata. Se la teoria supera il test viene confermata, altrimenti è sostituita da un'altra teoria falsificabile. Questo processo, evolutivistico, per congetture e confutazioni è ciò che garantisce il progredire della conoscenza. Si impara più dai fallimenti più che dalle conferme, poiché i fallimenti ci indicano dove le nostre teorie sono sbagliate e migliorabili.

Questo metodo basa anche la teoria politica alla base della Società Aperta (la società democratica, liberale, pluralistica e tollerante) dove non si interviene centralisticamente in modo top down sui fenomeni sociali, ma si correggono, secondo una "ingegneria sociale a spizzico", i problemi mediante interventi mirati, che possono essere corretti o abbandonati se si rivelano inefficaci. Ciò permette di apprendere dagli errori e adattarsi alle circostanze in continua evoluzione.

La **prototipation** e i processi che si fondano su di essa sono quelli, a livello epistemologico, più compatibili con l'innovazione tecnologica in una **società democratica e globalizzata**

Lo schema dell'epistemologia falsificazionista per trial and error

1. **P1** (Problema iniziale): Si parte da un problema o una domanda aperta che richiede una spiegazione.
 2. **TT** (Tentative Theory): Viene formulata una teoria o ipotesi come tentativo di risolvere il problema.
Questa teoria è una congettura che cerca di spiegare i fenomeni osservati.
 3. **EE** (Error Elimination): La teoria viene sottoposta a test critici, esperimenti e osservazioni. Se la teoria non supera i test, viene falsificata, ossia riconosciuta come errata.
Questo processo di falsificazione è l'elemento centrale
 4. **P2** (Nuovo Problema): Dalla falsificazione della teoria emerge un nuovo problema, o il problema originale rimane irrisolto, il che porta alla formulazione di una nuova teoria.
 5. Ripetizione del ciclo: Il processo si ripete con una nuova teoria (TT) che viene testata e, se necessario, eliminata attraverso l'eliminazione dell'errore (EE),
portando a un ulteriore problema (P2).
- Questa sequenza può essere formalizzata come segue:
 - ***P1→TT→EE→P2→ripetizione del ciclo***

INTRODURRE L'ETICA
NEI TEAM DI DESIGN:
L'ETICISTA DESIGNER

La pluralità di stakeholder caratterizzati da conoscenze (esplicite e tacite) e orizzonti valoriali diversi e la varietà di competenze oggi imprescindibili nel team di progettazionee necessitano una figura portatrice di skills etiche, epistemologiche e tecniche che faccia opera di chiarificazione e mediazione tra saperi e valori in vista di un design moralmente orientato: **l'eticista designer**: figura che incarna la svolta pragmatica dell'etica, e che agisce come value advocate per introdurre l'etica nella progettazione e implementazione di artefatti.

L'eticista designer può portare il suo peculiare contributo trovando modalità razionali di identificazione degli stakeholder e dei loro bisogni, di risoluzione di trade-off, conflitti e tensioni di valore che emergono durante i vari stadi del processo di design, suggerendo best practices e indicando potenziali rischi attingendo al suo bagaglio multidisciplinare di saperi e skills.

Questo non come esperto in senso tradizionale ma agendo come facilitatore, che porta una visione olistica al processo di design, che rende l'etica non una imposizione dall'alto, un mero washing morale o un ostacolo al team, ma un driver interno al processo di innovazione

L'eticista designer puà essere una figura chiave per garantire che il design non solo risponda alle necessità tecniche ed economiche, ma anche a quelle sociali, culturali e morali e che queste si alimentino l'una con l'altra, rispondendo anche alle richieste etiche imposte dalle normative nazionali e comunitarie

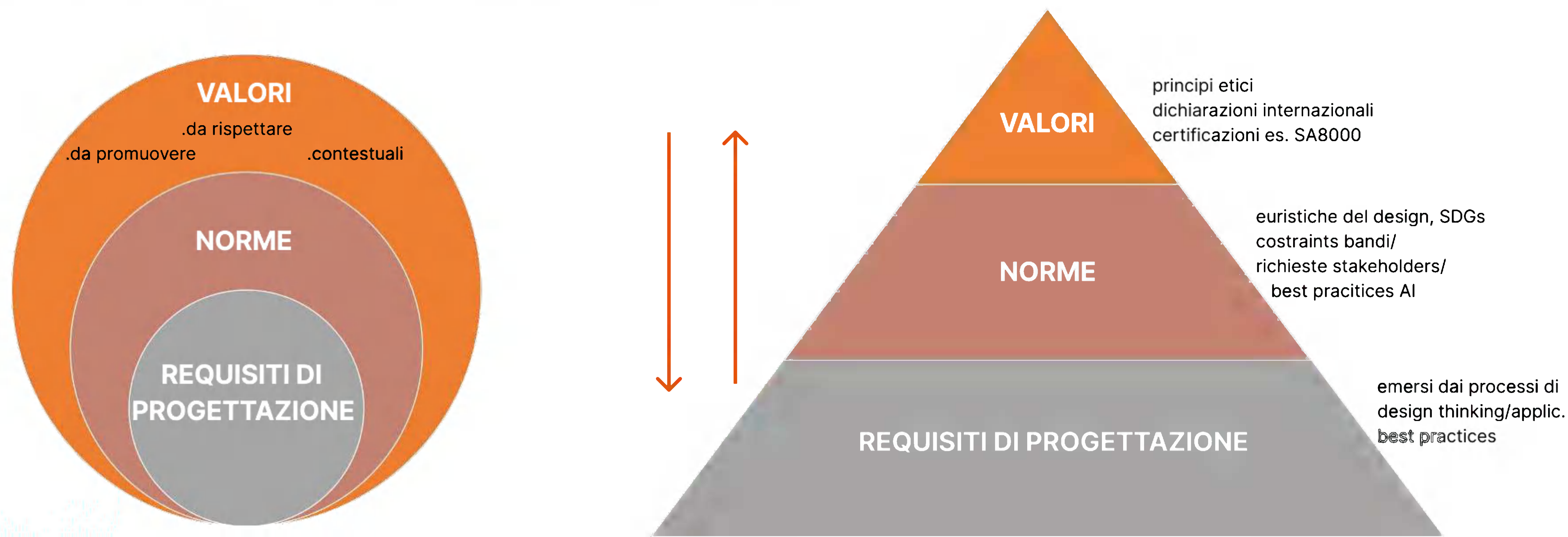
COMPITO ETICISTA DESIGNER	DESCRIZIONE
Identificazione degli Stakeholder	Identificare e coinvolgere una varietà eterogenea di stakeholder, inclusi quelli non immediatamente prevedibili, per assicurarsi che tutte le prospettive rilevanti siano considerate
Analisi della Conoscenza Tacita ed Esplicita	Valutare e chiarire la conoscenza tacita ed esplicita degli stakeholder per comprendere meglio i loro bisogni, valori e aspettative
Promozione della Multidisciplinarietà	Facilitare la collaborazione tra diversi esperti e discipline all'interno del team di design, garantendo che tutte le dimensioni del problema siano esplorate
Esplorazione delle Questioni Etiche	Identificare e chiarire le caratteristiche moralmente rilevanti della ricerca e del progetto, discutendo le implicazioni etiche delle scelte di design
Esplicitazione delle Scelte Etiche	Esplicitare le scelte etiche fatte dai designer e dagli altri membri del team, assicurando trasparenza e responsabilità nel processo di design
Ampliamento del Campo Prospettico del Team	Espandere la prospettiva del team includendo considerazioni etiche più ampie e coinvolgendo stakeholder al di fuori del gruppo immediato
Engagement con Teorie Metaetiche e Normative	Guidare il team nell'engagement con teorie metaetiche e normative, facilitando una comprensione profonda delle basi teoriche delle scelte etiche
Gestione dei Trade-off Valorial	Aiutare a identificare, discutere e risolvere i trade-off valoriali, facilitando decisioni che bilanciano i diversi interessi e valori coinvolti
Mediazione delle Tensioni Etiche	Mediare le tensioni etiche tra i membri del team e gli stakeholder, contribuendo a trovare soluzioni che siano moralmente accettabili e pratiche
Supporto nelle Decisioni di Progettazione	Assistere il team nella traduzione delle considerazioni etiche in concrete scelte di progettazione, assicurando che le decisioni finali siano allineate con i valori identificati.
Documentazione e Trasparenza	Garantire che tutte le decisioni etiche e i processi di deliberazione siano documentati in modo chiaro, mantenendo la trasparenza verso tutti gli stakeholder
Valutazione dell'Impatto Etico	Monitorare e valutare l'impatto etico delle decisioni di design nel tempo, suggerendo eventuali aggiustamenti o correzioni necessarie.

L'ETICISTA DESIGNER NEL TEAM: IL CONTRIBUTO ALLA TRADUZIONE DEI VALORI IN REQUISITI DI PROGETTAZIONE

Un compito in cui la figura dell'eticista designer può dare il suo contributo è quello della traduzione dei valori etici (beneficenza; non maleficenza; autonomia; giustizia ed esplicabilità,) coordinati con le euristiche, nei concreti requisiti di progettazione che emergono es. dal processo di design thinking

Questo processo non è una semplice deduzione dall'alto verso il basso o una mera applicazione di valori astratti, ma implica (dialetticamente):

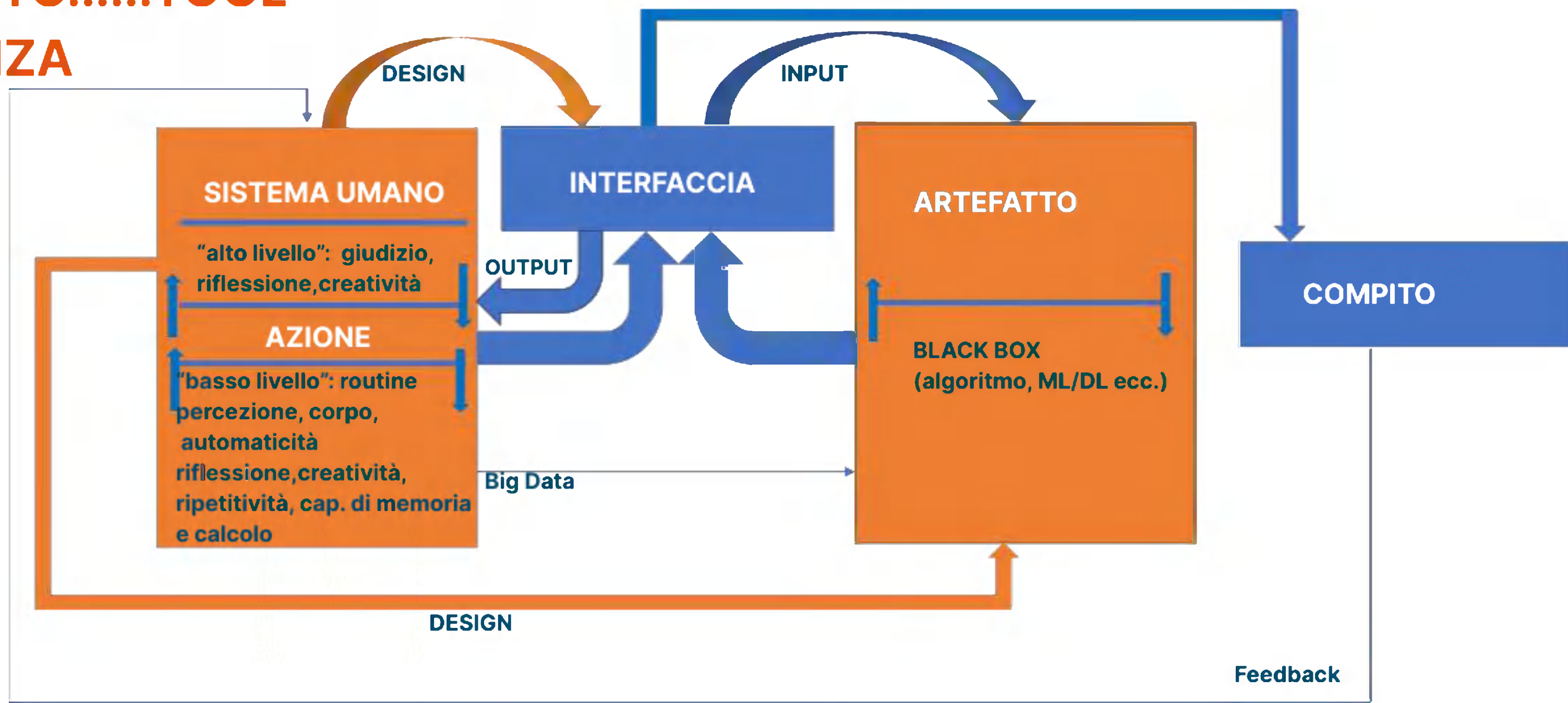
- Scoperta/integrazione del valore: (a) l'esplicitare i valori intesi come concepiti dagli ingegneri e dai progettisti; (b) scrutinare questi valori illustrando conflitti e scelte di valore; (c) confrontare i valori intesi nella letteratura etica. Es. ampliando i 5 valori etici AI-HCI con quelli che ispirano la policy etica dell'azienda o le diverse normative nazionali/internazionali o quelli di cui sono portatori i vari stakeholder
- (2) Traduzione dei valori nei requisiti di progettazione: (a) concettualizzazione del valore; (b) descrivere lo scollamento tra valori intesi e realizzati immaginando: best practices e cattive pratiche; usi e contesti non intenzionali dell'artefatto; (c) esplorare la traduzione / specificazione dei valori nei requisiti di progettazione e nelle varie affordances dell'artefatto



CONCLUSIONE: FROM TOOL TO.....TOOL

DALLA PAURA ALLA SPERANZA

- La rivoluzione digitale e delle ICT, e il sempre crescente potere computazionale a portata di ognuno, porta con sé aspettative messianiche e timori apocalittici (che inquinano il discorso pubblico sul tema)
 - Alcuni tra i rischi paventati, tralasciando quelli fantascientifici di una AI generale che sottomette l'umanità, sono di:
- Una **innovazione disruptive: crisi economica e perdita di massa posti di lavoro**
 - Crescita esponenziale delle disuguaglianze economiche e sociali tra gruppi e tra zone del pianeta
 - Atrofizzazione della capacità umana di giudizio, di scelta e delle skills creative delegate sempre più alla macchina



Questi rischi, reali, vanno affrontati innanzitutto a livello normativo capace di gestire l'accelerazione che le ICT causano nelle trasformazioni sociali. La es. rapida obsolescenza di alcune competenze e la richiesta di nuove va gestita politicamente e questa **transizione** è la sfida principale per le democrazie

Le nuove tecnologie sono *riserve ingegnerizzate della capacità di agire*: sono **strumenti** (eccellenti nei compiti complessi, incapaci in quelli difficili)

Il **design** secondo linee etiche diviene il modo, che affianca la dimensione normativa, di gestire nelle tecnologie stesse, e nel loro uso il potenziale, che offrono e volgerlo in modo positivo alla gestione della **complessità** → **INNOVAZIONE SOSTENIBILE**

La contemporaneità richiede l'uso di questi tools: è necessario acquisire le skills per utilizzarli e affrontare compiti sempre più complessi → cadono le dicotomie obsolete tra competenze: es. tra designer e sviluppatore

La progettazione, l'implementazione e l'uso etico e consapevole delle nuove tecnologie possono liberare le skills umane più elevate, disimpegnandole da compiti di basso livello ma di gran dispendio di risorse (cognitive, computazionali ecc.)

Conclusioni: Un ecosistema di informazioni, il *design* per un uso umano degli esseri umani

L'analisi condotta attraverso gli strumenti analitici e critici della storia della scienza e della riflessione filosofica ha mostrato come la relazione tra essere umano e macchina non sia il prodotto di un'evoluzione tecnica lineare o necessaria, ma l'esito di specifiche contingenze storiche, scientifiche e culturali. In questa prospettiva, la tesi ha individuato una direttrice di continuità capace di rendere intelligibile il passaggio che va dalla nascita della computazione e dalla formalizzazione dell'informazione fino alle forme contemporanee dell'Intelligenza Artificiale, mettendo in luce come concetti quali informazione, feedback, controllo, previsione e automazione abbiano progressivamente contribuito a ridefinire l'immagine dell'umano e le modalità dell'interazione con l'artificiale.

Attraverso gli strumenti interpretativi via via individuati nel corso della ricostruzione storico-critica, questa continuità consente di chiarire tanto i guadagni quanto le criticità delle tecnologie digitali contemporanee, rendendo visibili i limiti strutturali dei modelli informativi e statistico-predittivi, le conseguenze sociali della loro applicazione e lo scarto persistente tra computazione astratta e morfologia concreta dell'azione e della cognizione umana.

Al tempo stesso, il lavoro non si arresta sul piano teorico. I nodi emersi lungo questa linea di continuità vengono assunti come risorsa progettuale e tradotti in un modello operativo, concretizzato nel framework che chiude il lavoro, in cui il design è inteso come spazio di ricomposizione tra capacità computazionali e specificità cognitive, pratiche ed etiche dell'essere umano. In questo senso, risulta difficile - se non impossibile - comprendere come le tecnologie digitali possano essere orientate verso politiche inclusive o tradotte in best practices responsabili senza coglierne il *seme* originario, ossia le condizioni storico-culturali entro cui tali tecnologie sono state pensate, progettate e legittimate; solo riconoscendone il carattere intrinsecamente storico diventa possibile intervenire consapevolmente sulle modalità della loro implementazione odierna e sulla pervasività che esse assumono nell'ecosistema informativo contemporaneo.

Nel suo testo del 1950 *The Human Use of Human Beings* [Wiener, 1950], pubblicato soltanto due anni dopo *Cybernetics, or Control and Communication in the Animal and the Machine* [Wiener, 1948] – il testo in cui veniva “istituzionalizzata” la cibernetica come scienza capace di studiare ogni sistema, indipendentemente dal suo sostrato materiale (biologico o

artificiale) in quanto descrivibile come sistema informazionale a retroazione negativa - Norbert Wiener, preoccupato che gli esseri umani potessero essere “lasciati fuori dal feedback perché troppo lenti” [Numerico, 2021, p. 16], osservava riguardo allo sviluppo delle tecnologie che oggi segnano la contemporaneità, in modo quasi profetico, come:

“La società può essere compresa soltanto attraverso lo studio dei messaggi e dei mezzi di comunicazione relativi ad essi; e che, nello sviluppo futuro di questi messaggi e mezzi di comunicazione, i messaggi fra l'uomo e le macchine, fra le macchine e l'uomo, e fra macchine e macchine sono destinati ad avere una parte sempre più importante.” [Wiener, 1950, pp. 23-24, trad. it.]

Con queste parole, Wiener anticipava non solo la pervasività attuale di strumenti capaci di comunicare con l'uomo e tra di loro in modo sempre più autonomo, ma anche l'avvio di quella che sarebbe stata definita la *rivoluzione dell'informazione* [Bynum, 2010; Floridi, 2010]. Con essa nasce la tensione tecnica, etica e socio-culturale destinata a segnare ogni successiva interazione fra essere e macchina. Come osserva il filosofo della tecnologia e della scienza T. W. Bynum: “La rivoluzione dell'informazione [...] consiste proprio nel numero esponenzialmente crescente di trasformazioni sociali e di problemi etici resi possibili dalla tecnologia elettronica, dallo studio scientifico dell'informazione e dalla nascita della cibernetica,” nascita che si avvia con “gli sviluppi tecnici e scientifici degli anni Quaranta che resero possibile una straordinaria “esplosione” di nuovi prodotti e innovazioni negli anni Cinquanta e oltre.” [Bynum, 2010, trad. mia]¹

Queste considerazioni offrono la prospettiva in cui si sono intessuti gli interrogativi e i tentativi di risposta che hanno guidato il presente lavoro di ricerca, volto a rispondere ad una domanda di fondo: “*Stiamo cercando di creare un sostituto meccanico dell'essere umano, o uno strumento che lavori in simbiosi con l'uomo, un'estensione delle sue facoltà mentali?*” [Ceruzzi, 2012].

A partire da questo sfondo storico e concettuale, la trama e l'ordito di questa ricerca si sono avvolti come una spirale attorno a un concetto assunto come prisma critico per affrontare i molteplici interrogativi condensati nell'osservazione wieneriana e nell'interrogativo guida del presente lavoro: la metafora cibernetica, intesa come analogia funzionale tra sistemi biologici

¹ “The relevant technical and scientific innovations included the creation of electronic computers, the development of a scientific theory of information, and the birth of cybernetics. These mark the beginning of the Information Age, as I understand it, and they generated an enormous number of social and ethical changes and challenges. The Information Revolution, in my view, is the exponentially growing number of social changes and challenges enabled by electronic technology, the scientific study of information, and the birth of cybernetics”.

e artificiali, entrambi descrivibili attraverso meccanismi di feedback e analizzabili su una base comune - quella dell'information processing.

In questo quadro, il percorso di ricerca ha cercato di integrare in un quadro unitario i diversi livelli di analisi - storico-scientifico, tecnico, matematico e critico-filosofico - mostrando come la metafora cibernetica possa costituire la dorsale interpretativa dell'evoluzione del rapporto uomo-macchina e, insieme, la chiave per orientarne in senso umano e sostenibile le applicazioni contemporanee nel design *delle* tecnologie digitali

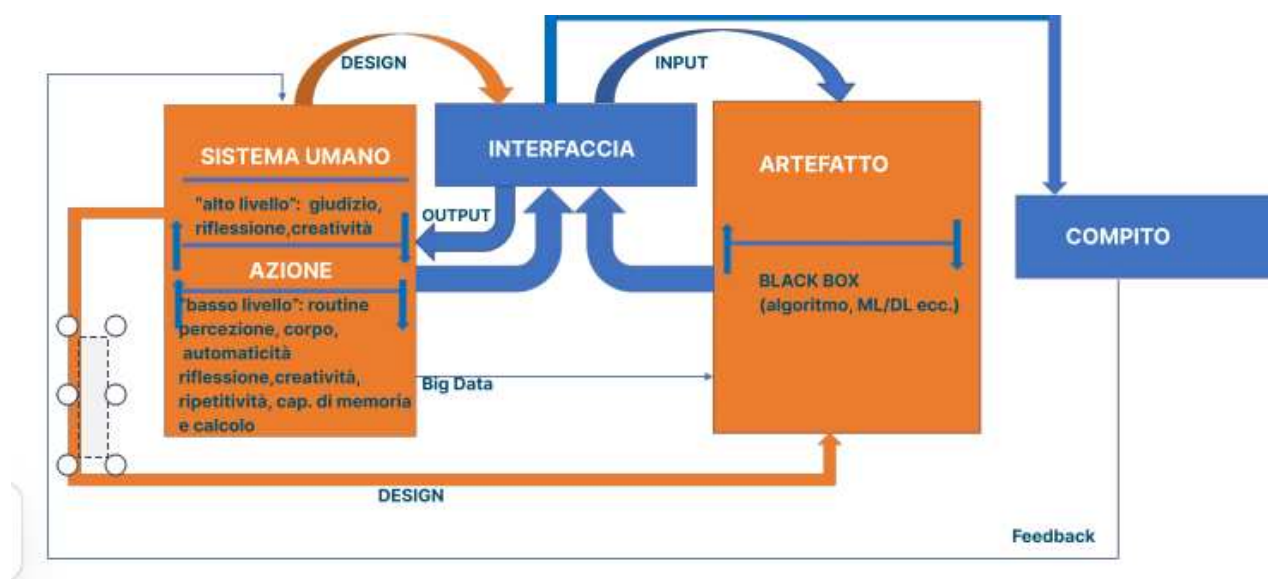
Attraverso tale approccio è stato possibile ricostruire la traduzione di concetti teorici in strumenti concreti, le condizioni tecniche della loro realizzazione e le ricadute culturali e sociali del paradigma cibernetico, che diventa così il filo conduttore della ricerca e lo strumento per leggere le trasformazioni dell'umano nell'era digitale.

L'impiego della metafora cibernetica si è infine rivelato particolarmente proficuo nel concepire la struttura d'interazioni ormai ubiqua tra esseri umani e sistemi artificiali, propria della continua mediazione tecnologica dell'azione nell'età dell'informazione. Lo schema [Fig. 31] rappresenta visivamente il modello di interazione uomo-macchina elaborato nell'ambito dell'attività applicativa di *design thinking* svolta presso Exprivia Spa. Esso traduce la metafora cibernetica in una configurazione sistemica di retroazione, nella quale il compito (*task*) costituisce il punto di chiusura del circuito informazionale. Il sistema umano è articolato su due livelli funzionali: un livello "alto", associato a giudizio, riflessione e creatività; e un livello "basso", relativo a percezione, automatismi, corpo, memoria e calcolo.

Entrambi interagiscono dinamicamente con l'artefatto, concepito come black box (algoritmo, machine learning, deep learning, ecc.), attraverso un'*interfaccia* che media il flusso di input e output. Nel processo complessivo, il compito funge da elemento regolativo: orienta l'interazione e definisce il ciclo di *feedback* e *feed-forward* che collega azione umana e risposta artificiale. L'interfaccia diventa così il luogo della co-costruzione cognitiva, in cui l'informazione viene continuamente scambiata, filtrata e rielaborata.

Lo schema mostra come, entro questo *ecosistema* informazionale, il design agisca sia a monte che a valle del processo, modellando la distribuzione delle funzioni cognitive tra uomo e macchina e chiudendo il circuito di retroazione. La metafora cibernetica, in questa prospettiva, si rivela quindi non solo un principio descrittivo, ma anche operativo: il dispositivo concettuale che consente di comprendere e progettare l'interazione come forma di co-adattamento tra umano e artefatto orientato al compito.

Fig. 31 Il modello di interazione uomo-macchina elaborato per questa tesi presso Exprivia Spa



Questo schema, tuttavia, nasce e acquista senso all'interno di precise coordinate concettuali e vettori di sviluppo storico, analizzati nel corso di questo lavoro. Proprio tali coordinate ne hanno reso possibile la costruzione e offrono le basi per orientarne il perfezionamento futuro, in direzione di un'interazione tra umano e macchina sempre più sostenibile e capace di valorizzare il polo umano della relazione.

L'analisi, allora, non poteva che aprirsi, nel primo capitolo, al livello di astrazione del *Whole System*: la visione sistemica da cui la cibernetica wieneriana ha preso forma e che, per la prima volta, ha posto uomo e macchina su un piano di equivalenza. Solo nelle condizioni sociotecniche della guerra - dove acciaio e carne divennero indistinguibili - è stato possibile concepire l'umano come parte di un circuito di retroazione, un servomeccanismo capace di correggersi attraverso il feedback dell'ambiente [Galison, 1994].

Il "personale sforzo bellico" di Norbert Wiener - il progetto del predittore antiaereo, finanziato con soli 2325 dollari - nacque come strumento per rendere più efficiente la capacità contraerea alleata di abbattere velivoli nazisti, ma attraverso l'analisi condotta in questo lavoro ne ha rivelato, una volta "universalizzata", la propria natura più profonda: non semplice dispositivo tecnico, bensì matrice di una nuova concezione dell'uomo e della natura come sistema di *comunicazione e controllo*. Ciò che era nato per prevedere le traiettorie di un aereo nemico divenne così la chiave per "comprendere l'intero sistema", secondo l'espressione che una rivista della controcultura americana degli anni Sessanta avrebbe poi assunto come motto

[Turner, 2006], e in cui le macchine cibernetiche vennero reinterpretate come amplificatrici della mente: un equivalente meccanico degli psichedelici, strumenti di espansione e potenziamento delle capacità cognitive umane.

Da quell'intuizione prende forma la nascita del paradigma uomo-macchina come sistema integrato di retroazione, fondato non più sulla distinzione tra soggetto e strumento, ma sulla loro continuità funzionale e informazionale.

L'analisi ha messo in luce, innanzitutto, il senso *unheimlich*, perturbante, che ha accompagnato la nascita della cibernetica e che continua a riflettersi nelle sue evoluzioni contemporanee. A turbare Stibitz e Weaver, i superiori di Wiener nell'apparato scientifico-militare statunitense che videro all'opera il predittore antiaereo, non fu la somiglianza materiale della macchina con l'essere umano o sue parti o caratteristiche [Cfr. Roselli et al., 2025], bensì la sua *funzione*: il comportamento, apparentemente animato, con cui essa imitava l'atto umano di prevedere e correggere. Ma questo senso di *uncanny* non fu un fraintendimento, bensì una spia delle condizioni di possibilità socioculturali che resero pensabile l'analogia tra uomo e calcolatore. Esso segnala il punto in cui la cultura tecnico-militare della guerra rese plausibile descrivere entrambi come sistemi di elaborazione dell'informazione regolati da principi di feedback.

Questa prospettiva ha messo in luce, dal punto di vista "cibernetico", la riduzione dell'uomo a *cluster* statistico di dati, che agisce omeostaticamente come sistema di retroazione negativa - le stesse criticità che il seme cibernetico ha fatto sbocciare nelle tecnologie digitali contemporanee. Le dinamiche di previsione, adattamento e correzione che definivano il predittore riemergono oggi nei sistemi di *machine* e *deep learning*: la continuità tra paradigma cibernetico e logica algoritmica si manifesta nella riduzione del reale a modelli generalizzabili, nella compressione della varietà umana in *cluster* statistici [Crawford, 2021], nella *black box society* [Pasquale, 2015] e nei *bias* strutturali che producono nuove disuguaglianze [O'Neil, 2016]. L'assunzione del *principio del mondo stabile* [Gigerenzer, 2022] consolida la ripetizione del passato nel presente, trasformando la temporalità stessa in un circuito chiuso di previsione e retroazione [Nowotny, 2022].

Ma il predittore di Wiener - una sorta di Hölderlin cibernetico - contiene già in sé una duplicità: il pericolo e ciò che salva. Nella sua struttura di retroazione risiede al tempo stesso la minaccia della riduzione e la possibilità di una liberazione, di una nuova alleanza tra uomo e informazione - un'ambivalenza che riecheggia nell'attuale dibattito, sempre oscillante tra l'apocalittico e il messianico, sulle tecnologie digitali.

Questa stessa ambivalenza era già riflessa nelle visioni opposte di Ken Kesey, Mario Savio, Richard Brautigan e Stewart Brand - la *Combine di Qualcuno volò sul nido del cuculo* e il *cybernetic meadow* di *All Watched Over by Machines of Loving Grace* - che segnano il passaggio dalla cibernetica di guerra alla controcultura americana, dal controllo sistemico alla ricerca di un'armonia simbiotica tra umano e macchina. In esse la cibernetica non appare più soltanto come scienza del comando e della previsione, ma come linguaggio capace di descrivere nuove forme di interazione, cooperazione e interdipendenza.

La cibernetica, come tecnica, metafora e retorica, resta così la matrice ambigua della modernità digitale: logica di riduzione e, insieme, desiderio di liberazione attraverso l'informazione, in un rapporto che, venute meno le differenze sostanziali tra sistemi artificiali e biologici, pone quest'ultima come livello ontologico minimo di ogni sistema, la chiave di una nuova ecologia dell'informazione.

Proprio la nascita della teoria rigorosa dell'informazione elaborata da Claude Shannon [Shannon, 1948] ha segnato un ulteriore passaggio chiave. In primo luogo, il matematico dei Bell Labs ha dato il via concettuale alla metafora interpretativa di questo lavoro, mostrando l'analogia tra la più semplice delle componenti elettromeccaniche - il relè, dispositivo in grado di assumere due stati discreti ("acceso/spento", "0/1") - e le funzioni logiche booleane.

In secondo luogo, la matematizzazione del concetto di informazione, nata nel contesto bellico della crittoanalisi [Shannon, 1945] e dalla necessità di distinguere il segnale utile dal rumore, mostra - come messo in luce nel presente lavoro - che il modello di comunicazione e la formalizzazione dell'informazione elaborati in quegli anni coincidono quasi integralmente con quelli presentati nel 1948. Ne risulta una visione in cui, nell'indifferenza tra i diversi sistemi di comunicazione, l'informazione, nel suo nesso con l'entropia, si configura come il livello ontologico minimo di ogni sistema, indipendentemente dal suo sostrato materiale.

La semantica viene completamente sacrificata a favore della quantità di informazione trasmessa, del grado di ordine o imprevedibilità che un messaggio può mantenere lungo il canale: ciò che conta non è più che cosa viene comunicato, ma quanto - e con quanta efficienza. L'informazione, in relazione matematica con l'incertezza, diviene così la misura di ogni processo comunicativo, biologico o artificiale, che può essere descritto e quantificato, comune a ogni sistema informazionale - umano, ibrido o artificiale - che può così essere messo in comunicazione.

La formula di Shannon $H = -\sum_{i=1}^n p_i \log p_i$ [Shannon, 1945; Shannon, 1948] [Shannon, 1945; Shannon, 1948] costituisce l'intelaiatura formalizzata del percorso che, dal predittore

wieneriano, conduce ai moderni sistemi integrati - dove le tecnologie digitali tendono a divenire parte dell'architettura cognitiva estesa umana [Clark, 2025] - nei quali il *design* delle interazioni è volto a ridurre l'entropia e a favorire l'armonizzazione dello scambio informativo tra polo umano e polo artificiale.

L'analisi ha mostrato che questo sistema di informazioni trova la sua plausibilità solo entro il quadro "cibernetico" precedentemente descritto e di una visione della mente e del cervello umano come sistemi di elaborazione dell'informazione digitale. In questa prospettiva si colloca il modello logico di Warren McCulloch e Walter Pitts [McCulloch, Pitts, 1943], che rappresenta il cervello come una rete di neuroni formali capace di simulare, attraverso connessioni logiche e soglie di attivazione, i processi del pensiero. Tale impostazione, che equipara la mente a una macchina logica, ha fornito il fondamento tecnico della computazione neurale, ma lascia aperta una domanda decisiva: questa logica, che costituisce la cornice teorica della considerazione culturale degli strumenti cibernetici come amplificatori delle capacità mentali, è sufficiente a rappresentare il pensiero umano nelle sue facoltà?

Analizzando, dal punto di vista storico e teorico, le analogie e le differenze del modello "Wiener-Shannon" [Wiener, 1956], emerge con chiarezza un punto critico: nella maggiore attenzione data da Wiener al contesto e alla dimensione semantica riappare, nel senso più letterale, la carne del cervello umano. A un livello d'astrazione più elevato [Marconi, 2025] - logico-matematico e di teoria della computabilità - questa dinamica riemerge nell'analisi del lavoro di Alan Turing, considerato nel secondo capitolo.

L'analisi del secondo capitolo ha mostrato come, nel pensiero di Alan Turing, la riflessione sui limiti e le potenzialità della computazione costituisca un punto di svolta per comprendere il rapporto tra mente e macchina. L'indagine sul lavoro del matematico inglese è stata particolarmente utile ad esaminare la dorsale di *imitazione - riduzione-integrazione/blurring* umano/macchina - *riproduzione/reificazione* che struttura la storia del rapporto uomo-macchina dal livello di analisi logico-formale.

Sin da *On Computable Numbers, with an Application to the Entscheidungsproblem* [Turing, 1936] e, successivamente, nella tesi di dottorato *Systems of Logic Based on Ordinals* [Turing, 1939], emerge una tensione tra *intuizione ingegno* [Turing, 1939; Hodges, 1985], segno di un residuo irriducibile, costante tra la manipolazione sintattica dei simboli - propria delle macchine universali - e l'aspirazione a riprodurre le facoltà cognitive del cervello umano. "" Il nostro compito è scoprire la natura di questo «residuo» e cercare di «copiarlo» nelle macchine" [Turing, 1948]

Di questa tensione sono segno il lavoro sulle *macchine non organizzate* - le *child-machines* - e l'*Imitation Game* [Turing, 1948; Turing, 1950], analizzati in dettaglio in questo lavoro. In essi Turing tenta di tradurre e mettere alla prova, in sostrati materiali diversi, le capacità di apprendimento, memoria e adattamento proprie dell'intelligenza, ossia la descrizione computazionale del funzionamento cerebrale. Come egli stesso scrisse a W. Ross Ashby, il suo obiettivo era "costruire un cervello": riprodurre, attraverso un modello meccanico, i principi organizzativi dell'intelligenza naturale. In tal senso, tali esperimenti mostrano insieme la portata e il limite della riproducibilità dell'intelligenza: la mente può essere formalmente descritta, ma non integralmente ridotta al calcolo.

L'analisi ha chiarito che, nelle tesi di Turing e Church sulla computabilità, il calcolo non rappresenta soltanto una capacità, ma anche un confine: una scoperta circa le limitazioni del potere formalizzante dell'*Homo sapiens* [Post, 1936]. Essa ha inoltre permesso di mettere in luce, al livello logico-computazionale, la duplice eredità che le ricerche di Turing hanno lasciato all'intelligenza artificiale: da un lato il paradigma simbolico dell'intelligenza artificiale "classica", la GOF AI [Haugeland, 1985], che concepisce la mente come un sistema di manipolazione di simboli secondo regole esplicite e formalizzabili [Newell, Simon, 1976]; dall'altro il paradigma connessionista, anticipato dal suo lavoro sulle macchine non organizzate, che interpreta l'intelligenza come risultato di processi distribuiti e auto-organizzativi, modellabili attraverso reti neurali artificiali [Copeland, Proudfoot 1996 e 1999; Copeland 2004cc, pp. 403-409; Dewhurst, 2019].

La *scarnificazione* dell'essere umano impegnato in un compito di calcolo - nucleo concettuale della macchina di Turing e base per la definizione rigorosa di algoritmo - ha messo in evidenza la *simbiosi* sostanziale tra procedura algoritmica ed essere umano. La macchina di Turing presuppone infatti la presenza di un operatore umano come parte integrante del sistema: è l'essere umano a formulare le domande, fornire le risorse materiali e interpretare le risposte [Dodig-Crnkovic, 2011].

Come ridare corpo - un corpo metallico - a queste idee, e in particolare a macchine che vedono nella memoria la chiave per riprodurre l'intelligenza umana e per regolare i rapporti con la controparte biologica a cui la loro architettura si ispira? Dato il sapore decisamente industriale del suo resoconto delle macchine - con i riferimenti non solo al nastro di carta perforato, ma persino a ruote e leve all'interno dello *scanner* - appare evidente come Turing concepisse l'intelligenza come un processo capace di essere meccanizzato, reso concreto nel ritmo dei suoi organi di memoria e di controllo [Cfr. Copeland, 2017, p. 54].

L'analisi, condotta nella parte centrale di questo lavoro - di natura storico-scientifica e storico-tecnologica - ha messo in evidenza, come indicato dalla storiografia della computazione [Ceruzzi, 2003; Ceruzzi, 2012; Haigh, Ceruzzi, 2021], che gli anni Quaranta rappresentarono un decennio cruciale, in cui si iniziò a riflettere in modo profondo sul ruolo dell'essere umano nel rapporto con le macchine di controllo complesse [Cfr. Ceruzzi, 2012, p. 42], secondo quattro direttrici di sviluppo che segnarono l'evoluzione del calcolo automatico e delle sue applicazioni, che possiamo qui riassumere in relazione a quanto preso in esame: *il paradigma digitale; la convergenza tecnologica; l'elettronica a stato solido e interfaccia uomo-macchina*.

L'indagine ha mostrato che il passaggio dall'analogico al digitale non fu una semplice innovazione tecnica, ma una trasformazione concettuale e materiale che ridefinì il modo di concepire informazione, calcolo e mente. Le macchine analogiche, come il *Differential Analyzer* di Vannevar Bush, basate sulla rappresentazione continua delle grandezze fisiche, riproducevano meccanicamente le relazioni matematiche, ma la loro precisione era limitata da attriti e instabilità, rendendole inadeguate alla crescente domanda di rapidità e affidabilità imposta dal contesto bellico.

Fu questa esigenza sociotecnica a determinare la nascita del *paradigma digitale*, che sostituì la continuità fisica con la discrezione logica, traducendo il reale in sequenze binarie: acceso/spento, 0/1, vero/falso. I simboli della macchina di Turing, i valori booleani del calcolo logico e gli stati elettrici dei circuiti costituirono la base di un linguaggio universale capace di unificare logica, fisica e mente entro una stessa logica dell'informazione.

A *Bletchley Park*, le macchine *Heath Robinson* e *Colossus*, rimaste segretate per decenni, resero operativa questa logica: i nastri perforati del messaggio e delle chiavi, sincronizzati meccanicamente, automatizzarono i metodi statistici di Turing e Tutte per la decifrazione dei cifrari *Tunny*. Le valvole termoioniche, equivalenti funzionali dei neuroni, aprivano e chiudevano i circuiti alla velocità dell'impulso elettrico, dando corpo alla logica binaria. Negli Stati Uniti, l'*ENIAC* - primo grande calcolatore elettronico *general purpose* - applicò lo stesso principio al calcolo numerico, eseguendo migliaia di operazioni al secondo. La programmazione, affidata alle operatrici che configuravano manualmente pannelli e cavi, mostrava tuttavia quanto l'automazione elettronica dipendesse ancora dall'intervento umano, prefigurando la futura interfaccia uomo-macchina.

La *convergenza tecnologica* si è mostrata come il risultato diretto del paradigma digitale, che mette in comunicazione discipline fino ad allora autonome - logica matematica, teoria dell'informazione, elettronica e neurofisiologia - unificandole entro un medesimo orizzonte

tecnico. La guerra ne ha costituito il laboratorio, fondendo calcolo teorico e costruzione ingegneristica con quest'ultima che è stata presa in esame nel dettaglio sotto il punto di vista delle prestazioni e delle potenzialità computazionali in tutte le macchine analizzate.

. A Bletchley Park, la collaborazione tra matematici, tecnici e operatrici nella progettazione di Colossus rende possibile l'automatizzazione dei metodi statistici di analisi; sull'altra sponda dell'Atlantico, l'ENIAC concretizza l'integrazione fra logica simbolica, fisica dei circuiti e lavoro operativo. La convergenza tecnologica, resa possibile dalla comune matrice digitale, segna così la nascita della scienza computazionale come campo collettivo e interdisciplinare, in cui teoria, materiali e cognizione umana si saldano entro un unico sistema informazionale.

La terza direttrice, quella dell'*elettronica a stato solido* a mostrato come la metafora cibernetica abbia superato i limiti dell'implementazione elettromeccanica per viaggiare e incarnarsi a *velocità elettronica*. L'impiego delle valvole termoioniche (i tubi a vuoto) e dei tubi catodici nelle prime macchine come Colossus o il *Manchester Baby* in Inghilterra o nell'ENIAC negli USA ha permesso di sviluppare capacità computazionali sempre maggiori, di superare i limiti delle leggi della fisica classica newtoniana. L'invenzione del transistor da parte di Schokleu, Brattain e Bardeen nel 1948 e quella del circuito integrato (IC) di Noyce e Kilby a fine anni '50, e la loro progressiva diffusione, ha avviato una rivoluzione verso la miniaturizzazione e la personalizzazione che ha dato corpo *in hardware* alle speranze controculturaliste che vedevano nei computer strumenti di espansione, come tecnologie *small scale*, sui dispositivi cibernetici.

La quarta direttrice, quella dell'*interfaccia uomo-macchina*, ha messo in luce come già nei calcolatori degli anni Quaranta fosse presente, *in nuce*, una relazione *simbiotica* tra l'operatrice e la struttura del calcolatore. Nella macchina di Flowers era presente soltanto una memoria fisica per "programmi potenziali": senza l'essere umano impegnato a cablare i pannelli, restavano soltanto prese vuote e interruttori inerti [Wells, 2006]. Le funzioni che Turing e von Neumann, altro grande protagonista della nostra storia, intendevano incorporare nelle macchine - memoria, controllo, adattamento - rimanevano ancora delegate all'azione umana. *Colossus* forniva un supporto quantitativo alla decrittazione, ma il suo impiego anticipava una delle qualità fondamentali dei computer moderni: la profonda interazione con l'essere umano. L'integrazione fra analisti, Wrens e macchine costituiva un unico sistema operativo distribuito, in cui la componente umana fungeva da mediatore cognitivo e operativo, da "processore esterno" del calcolo elettronico.

Parallelamente, con l'ENIAC, il passaggio dall'essere umano, spesso una donna, computatore (*computer*) alla macchina computazionale (*computer*) [Mauchly, 1942], come l'analisi ha mostrato, ha reso esplicito questo cambiamento di ruolo: le matematiche, un tempo impegnate nei calcoli "*pencil and paper*", si trasformarono nelle operatrici che configuravano i cavi, i pannelli e le connessioni logiche della macchina. L'elemento umano non era totalmente sostituito ma veniva incorporato nel sistema tecnico, come gestore della complessità e interfaccia attiva tra il linguaggio logico della macchina e i compiti specifici assegnati. In questa interazione strettissima tra calcolatore e operatrice si delineava già la forma primitiva della simbiosi uomo-macchina, che avrebbe definito l'evoluzione successiva delle tecnologie digitali.

Lo stupore profondo che gli operatori provano di fronte a *Colossus* [Good, Michie, Timms, 1945, 51(j), p. 327, trad. mia], così come la reazione "mediatica" suscitata dal display luminoso dell'*ENIAC*, che porta la stampa a parlare di "cervelli giganti" [Thielmann, 2019; Martin, 1995], testimoniano l'impatto che anche le più imponenti e primitive macchine computazionali degli anni Quaranta esercitano sulla riflessione riguardante il rapporto tra uomo e calcolatore, tra antropomorfizzazione tecnologica e ripensamento del sé.

L'evoluzione delineata dalle quattro direttrici ha trovato la propria sintesi nel principio del programma memorizzato, la più compiuta incarnazione tecnica delle idee di Turing sui calcolatori universali. La descrizione dell'architettura di von Neumann, redatta a partire dagli studi di Mauchly ed Eckert per il progetto EDVAC, rappresentò la concretizzazione di un modello in cui dati e istruzioni condividevano la stessa memoria: un organismo unico capace di leggere, scrivere e modificare sé stesso.

La struttura dell'EDVAC, con i suoi organi funzionali - memoria, unità aritmetico-logica, controllo, input e output - riproduceva la suddivisione biologica di un organismo, secondo il linguaggio allora dominante della cibernetica. La descrizione dell'architettura di von Neumann ne formalizzava il funzionamento in termini logici e sequenziali, introducendo il concetto di ciclo di istruzione e di elaborazione seriale dei dati come principio organizzativo dell'intero sistema.

In questa prospettiva, l'EDVAC divenne il paradigma da cui si irradiò l'intera genealogia del calcolo moderno. Il principio del programma memorizzato ha trasformato la macchina da strumento operativo a sistema logico integrato, in cui memoria e istruzioni si riflettono reciprocamente, segnando il punto in cui la metafora cibernetica trovò la propria piena realizzazione.

Ma l'analisi non poteva non interrogarsi su un punto: la serialità dell'architettura di von Neumann è davvero il modello più adeguato per esprimere la metafora cibernetica? L'esame dei colli di bottiglia logici e fisici della sua architettura - derivata dalle teorie di Turing e dai modelli neuronali di McCulloch e Pitts - ha aperto la strada a una sorta di *critica della cibernetica su se stessa*, mettendo in discussione la possibilità di ridurre la complessità biologica e cognitiva degli "automi" computazionali - artificiali o naturali - a una sequenza lineare e gerarchica di operazioni.

Prima di riassumere i risultati ottenuti in questa direzione, il lavoro non poteva prescindere, nel quinto capitolo, dal considerare - dato il carattere sociotecnico delle tecnologie analizzate [Johnson, Watmore, 2021] - gli effetti socioculturali dell'espansione del calcolatore e della *metafora cibernetica* che ne ha orientato la progettazione e la retorica, contribuendo, in modo tutt'altro che "neutro" [Sismondo, 2010], alla loro propagazione.

L'affermazione del "paradigma EDVAC" [Haigh et al., 2016] negli anni Cinquanta e Sessanta segnò il passaggio dal calcolatore come strumento sperimentale a infrastruttura scientifica, commerciale e militare, dominata dal monopolio IBM. Progetti come il SAGE (*Semi-Automatic Ground Environment*) - rete di difesa aerea che integrava radar, calcolatori IBM e operatori umani - mostrarono come il computer potesse divenire principio organizzativo dell'intera società. In SAGE, la metafora biologica della retroazione trovava piena attuazione: il sistema funzionava come un grande organismo biologico di *information processing*, espressione della logica del "mondo chiuso" della Guerra Fredda [Edwards, 1996].

La cibernetica divenne così linguaggio comune per descrivere tanto i processi sociali quanto quelli tecnici: da un lato, nella sociologia sistemica di Talcott Parsons, modello di equilibrio e autoregolazione; dall'altro, nella visione di Norbert Wiener, strumento di cooperazione e libertà comunicativa. Negli anni Sessanta, lo sviluppo dell'elettronica a stato solido e del microprocessore (Hoff, Faggin, Mazor, Intel, 1971) rese possibile la miniaturizzazione dei circuiti e la progressiva diffusione dei calcolatori concorrenti all'IBM, come i PDP della Digital Equipment Corporation, sempre più personali. Il passo da calcolatori colossali al *personal computer* - che avrebbe trovato piena realizzazione a partire dalla seconda metà degli anni Settanta con l'Apple I (1976) e l'IBM PC (1981) - si stava per compiere.

Il passaggio dal mainframe al time-sharing, introdotto nei primi anni Sessanta nei laboratori del MIT e del Project MAC (1963), segnò un cambiamento di paradigma: il calcolatore non era più una macchina centrale da servire, ma un dispositivo con cui interagire in tempo reale. È in questo contesto che J. C. R. Licklider formulò, nel 1960, l'idea di *man-computer*

symbiosis: la possibilità che uomo e macchina cooperassero, in piena coerenza con la metafora biologica della cibernetica, in un unico sistema cognitivo, ciascuno valorizzando le proprie capacità - l'intuizione e la decisione da un lato, la velocità di calcolo dall'altro.

In questo rovesciamento, il determinismo tecnologico risultava smentito: la storia della computazione non seguiva un'evoluzione autonoma, ma rifletteva scelte sociali e progettuali che reindirizzavano la metafora cibernetica dalla logica del controllo verso quella della collaborazione cognitiva e del calcolatore come strumento di comunicazione. ARPANET, l'antenata del moderno web, vide la luce proprio in questi anni, con la connessione del primo nodo tra UCLA e Stanford Research Institute nel 1969.

La computazione personalizzata attuale - dal personal computer agli smartphone, fino alle architetture di cloud e intelligenza artificiale distribuita - non può essere compresa se non alla luce di queste direttrici di sviluppo. Essa rappresenta l'esito di una lunga traiettoria che ha trasformato il calcolatore da macchina centrale di una struttura gerarchica, in cui l'essere umano era una delle tante informazioni da elaborare, a dispositivo di mediazione cognitiva, comunicativa e sociale.

Come l'analisi ha mostrato, se SAGE mirava a proteggerci dal primo fardello - la paura, quella atomica e comunista -, Grace Hopper volle liberarci dal secondo: quello della matematica. Il suo obiettivo era sollevare i matematici e le matematiche dal peso dei calcoli e della scrittura di istruzioni in linguaggio macchina, permettendo loro di dedicarsi ai problemi logici e concettuali. Con l'invenzione del primo *compilatore* (1952) e la progettazione del linguaggio COBOL, rese possibile tradurre comandi espressi in linguaggio naturale in istruzioni comprensibili per il calcolatore. Da questa svolta prese avvio la genealogia dei linguaggi di programmazione e dei sistemi operativi, che la nostra analisi non poteva tralasciare: un'evoluzione che, dal codice binario, ha progressivamente condotto a forme di comunicazione sempre più prossime al linguaggio umano, fino agli attuali modelli linguistici capaci di dialogare con l'uomo nel suo stesso idioma.

In questo percorso non si potevano tralasciare le conseguenze non solo epistemiche ma addirittura ontologiche dell'implementazione del *metodo Monte Carlo*, necessario ai calcoli per la bomba all'idrogeno di Teller, che hanno fatto detonare un nuovo universo: quello della *simulazione*. Dalla collaborazione fra Ulam, Metropolis e von Neumann nacque non soltanto una tecnica numerica, ma una nuova forma di conoscenza: in essa il computer cessava di essere mero strumento di calcolo per divenire elemento costitutivo di una realtà generata, un laboratorio virtuale in cui teoria ed esperimento si fondevano. L'implementazione del metodo

Monte Carlo sull'ENIAC - e poi sul MANIAC di Los Alamos - rappresentò, come mostrato, il passaggio dalla rappresentazione del mondo alla sua costruzione digitale: un nuovo modo non solo di conoscere la realtà, ma di abitarla e addirittura costruirla.

In questa prospettiva, il metodo Monte Carlo divenne anche uno spazio di incontro tra linguaggi e discipline eterogenei, una *trading zone* dove fisici, matematici e ingegneri svilupparono un *pidgin* tecnico condiviso capace di fondere lessici e pratiche differenti [Galison, 1996]. Da questa ibridazione prese forma una nuova cultura della progettazione, in cui la figura del *designer* assunse un ruolo centrale nella modellazione e nell'interazione con sistemi sempre più complessi. Queste dinamiche, sviluppatesi all'incrocio tra calcolo, rappresentazione e simulazione, prefigurano l'attuale integrazione uomo-macchina, in cui il confine tra reale e virtuale risulta sempre più sfumato e poroso. L'analisi del metodo Monte Carlo ha offerto a questo lavoro il destro per indagare gli elementi costitutivi dell'universo in questione: i *dati*. L'analisi del *metodo Monte Carlo* ha offerto a questo lavoro il destro per indagare gli elementi costitutivi dell'universo in questione: i dati.

A partire dalle tre "V" individuate da Doug Laney, *Volume, Variety, Velocity* [Laney, 2001], la crescita esponenziale della produzione e della circolazione dei dati ha dato origine a un nuovo regime epistemico fondato sulla misurazione e sulla predizione. Alle prime tre dimensioni si sono progressivamente aggiunte *Veracity* e *Value*, e via via ulteriori V, che hanno introdotto l'attenzione alle questioni di attendibilità, significato e valore ed eticità dei dati stessi.

In questa prospettiva, l'universo dei *big data* appare come l'erede diretto delle prime simulazioni Monte Carlo: una modalità di conoscenza che integra la probabilità e l'incertezza entro un sistema di controllo informazionale. L'analisi di questa evoluzione si lega e completa le riflessioni critiche e filosofiche sul concetto di *informazione* e di *dato* [Floridi, 2011, Floridi, 2012] sviluppate nei capitoli terzo e quarto, mostrando la continuità tra la dimensione teorica della cibernetica e le pratiche di elaborazione e interpretazione dei dati che caratterizzano la contemporaneità.

L'esame delle direttrici di sviluppo che condussero alla nascita dei primi calcolatori, giunte a compimento con l'"avvento" del paradigma EDVAC [Haigh et al., 2016; Dyson, 2012], ci ha lasciato con una domanda aperta riguardo alcune criticità insite nella stessa architettura proposta da John von Neumann.

I cosiddetti *bottlenecks* - o "colli di bottiglia" - rappresentano il limite strutturale derivante dal fatto che dati e istruzioni condividono un unico canale di comunicazione tra memoria e unità centrale di elaborazione, imponendo un flusso sequenziale che ostacola il

parallelismo e riduce l'efficienza del calcolo [Backus, 1978; Edwards & O'Keefe, 2016; Zou et al., 2021]. Tali *bottlenecks* costituivano, e costituiscono non solo un vincolo tecnico ma un vero e proprio ostacolo epistemologico nella mente del programmatore, poiché ne condizionano il modo stesso di concepire il calcolo e l'organizzazione della conoscenza computazionale.

Ci si può allora domandare, come fatto si è fatto in questo lavoro, se la *serialità* - principio strutturale dell'*architettura di von Neumann* - rappresenti un criterio di *design* adeguato per incarnare la metafora cibernetica che intendeva unificare mente e macchina. Bisogna chiedersi, in altre parole, se un modello fondato su una sequenza lineare di istruzioni può davvero esprimere la complessità dell'oggetto che pretende di riprodurre, o se, al contrario, la fedeltà alla logica biologista della cibernetica richiede, al contrario, un ripensamento radicale della forma stessa della computazione e del design che la realizza.

A fornire la risposta è lo stesso "padre" del paradigma EDVAC e dell'architettura eponima, l'uomo che più di tutti ha incarnato il ponte tra dimensione teorica, implementativa e sociotecnica dello sviluppo del calcolatore: John von Neumann.

Nel lavoro si è presa in esame una lunga lettera inviata a Norbert Wiener il 29 novembre 1946, nella quale il matematico ungherese esprime le sue perplessità sull'adeguatezza del modello logico di McCulloch e Pitts e di quello matematico di Turing - pur riconoscendone i meriti - a rappresentare la complessità del sistema nervoso. Osserva che i processi biologici non seguono gli schemi logici di queste formulazioni e che la loro stabilità dipende da ridondanza, errore e adattamento, non da componenti uniformi e deterministici.

Per affrontare il problema della complessità organizzativa, propone di passare dallo studio dell'elemento macro (il sistema nervoso o le sue parti) a quello micro, rivolgendosi a organismi più semplici come i batteriofagi, nei quali i meccanismi di auto-riproduzione e di organizzazione risultano più leggibili e passibili di una trattazione matematica rigorosa sul modello di Turing.

Da questa analisi emerge l'esigenza di una nuova teoria degli automi, fondata su probabilità, ridondanza e stabilità statistica, che anticipa le successive riflessioni di von Neumann sull'auto-riproduzione e sulla complessità dei sistemi naturali e artificiali. Il capitolo sesto di questo lavoro ha avuto per oggetto proprio la teoria degli automi di von Neumann.

Il capitolo ha mostrato come, nel passaggio dalla cibernetica "classica" alla teoria degli automi, von Neumann trasformi la metafora cibernetica dall'interno. Egli introduce una logica fondata su probabilità, ridondanza e stabilità statistica, integrando nel cuore della computazione i principi della teoria dell'informazione di Shannon. L'errore, da elemento

negativo, diventa parte costitutiva del funzionamento di ogni sistema informazionale, naturale o artificiale. La logica formale si apre così a una dimensione termodinamica e probabilistica, in cui l'affidabilità è misurata non in termini assoluti ma statistici.

In *The Computer and the Brain* (1958) questa trasformazione raggiunge il suo compimento. Il confronto tra gli automi computazionali per eccellenza, *cervello* e *calcolatore*, mostra che il primo opera secondo processi paralleli e parzialmente analogici, mentre il secondo rimane vincolato alla serialità e alla rigidità dell'architettura a programma memorizzato. Il cervello appare come un sistema ibrido analogico-digitale, la cui efficacia risiede nella combinazione di segnali continui e discreti e nella capacità di adattarsi all'errore, sfruttandolo come risorsa informazionale.

Da questa analisi emerge che la logica computazionale deve evolvere da uno schema combinatorio a uno probabilistico e neuromorfico - mostrando ancora una volta come l'integrazione nella logica formale-digitale della statistica informazionale faccia riemergere la "carne" del sistema nervoso - capace di rappresentare la varietà e la flessibilità dei processi cognitivi. von Neumann riconosce che la mente non può essere descritta come una sequenza lineare di istruzioni, ma come un intreccio dinamico di connessioni e adattamenti, in grado di operare anche in condizioni di errore e di bassa precisione.

L'analisi ha mostrato così che la riformulazione vonneumanniana della cibernetica - dal modello seriale del calcolatore a una concezione informazionale più articolata - ha aperto la strada a una visione della computazione come processo *parallelo*, *probabilistico* e *storico*, e *bottom-up*, anticipando i principi che saranno alla base del paradigma *connessionista* e delle moderne scienze cognitive, nelle quali il modello di mente non è più il calcolatore, ma il cervello stesso [Clark, 1989].

Il capitolo ha mostrato come, riconosciuti i limiti epistemici della serialità dell'architettura di von Neumann, la ricerca si sia orientata verso modelli di calcolo distribuito in grado di tradurre la svolta neuromorfica in termini sperimentali e implementativi.

Il Perceptron di Frank Rosenblatt [1958] costituisce il primo passo in questa direzione: una macchina capace di apprendere attraverso l'adattamento progressivo dei pesi sinaptici, anticipando la possibilità di un apprendimento autonomo della macchina migliorato rispetto ai modelli come quelli *machine learning* di Samuel sul gioco della dama [Samuel, 1959]. A esso si affiancano altri dispositivi come il *Pandemonium* di Oliver Selfridge (1959) e l'*Adaline* di Widrow e Hoff (1960), che esplorano la rappresentazione distribuita e il principio di apprendimento per rinforzo, che sono stati analizzati in questo lavoro.

Le critiche di Minsky e Papert [1969], che ne misero in luce le limitazioni logiche, non chiudono la ricerca – pur segnando un “inverno” per l’IA connessionista - ma ne hanno determinato una svolta metodologica. Negli anni Ottanta, con l’introduzione dell’algoritmo di *backpropagation*, l’apprendimento non è più inteso come semplice associazione diretta tra input e output, ma come processo di propagazione dell’errore all’indietro nei vari strati della rete, consentendo l’ottimizzazione progressiva dei pesi interni.

Il paradigma del *Parallel Distributed Processing (PDP)* [McClelland e Rumelhart, 1986] ha tradotto, come messo in evidenza, questi principi in un quadro teorico unitario: la mente e la macchina sono descritte come reti di unità semplici interconnesse, in cui la conoscenza non è localizzata, ma emerge dalla distribuzione delle attivazioni e dall’adattamento collettivo della rete.

In questa prospettiva, modelli e macchine come il Perceptron, Adaline, Pandemonium e le reti PDP segnano il passaggio dalla logica del calcolo deterministico a una computazione probabilistica e parallela, in cui la capacità di apprendere e generalizzare deriva dalla cooperazione di molte unità elementari. L’analisi ha messo così in evidenza come la metafora cibernetica, reinterpretata in chiave distribuita, prepari il terreno all’intelligenza artificiale connessionista e alla concezione contemporanea della computazione come processo adattivo e interattivo, di cui si sono esaminati alcuni passaggi fondamentali.

Dalla rinascita delle reti neurali negli anni Ottanta, con la *backpropagation* e il paradigma del *Parallel Distributed Processing (PDP)*, fino alla rivoluzione del *deep learning*, la ricerca ha progressivamente reso operative le intuizioni neuromorfiche che attraversavano la cibernetica. Le reti multilivello hanno permesso di rappresentare gerarchie di astrazioni e di apprendere direttamente dai dati, sostituendo la programmazione esplicita con l’auto-organizzazione delle connessioni interne.

Con i lavori di Geoffrey Hinton, Yann LeCun e Yoshua Bengio - i tre “*godfathers*” dell’intelligenza artificiale contemporanea - il connessionismo ha raggiunto una fase di piena maturità. L’introduzione delle reti convoluzionali per l’elaborazione visiva, il pre-training gerarchico dei modelli profondi e l’impiego congiunto di grandi quantità di dati e potenza di calcolo hanno portato al successo di AlexNet [2012], che segna una soglia decisiva nello sviluppo del *deep learning* e nel ritorno delle reti neurali al centro della ricerca scientifica e industriale.

Da questa soglia, le applicazioni si sono estese a domini sempre più complessi: dalla visione artificiale al linguaggio naturale, fino ai sistemi capaci di auto-apprendimento per

rinforzo, come *AlphaGo* [2016] e *AlphaZero* [2017]. La vittoria di *AlphaGo* sul campione mondiale di Go, Lee Sedol, rappresenta un momento simbolico persino più significativo di quello di *Deep Blue* su Kasparov vent'anni prima: non la semplice esecuzione di calcoli, ma la scoperta autonoma di strategie, la capacità di apprendere da sé, senza istruzioni esplicite, in un gioco dalla complessità incommensurabilmente superiore agli scacchi.

Il capitolo ha mostrato così come il percorso che va dal Perceptron al deep learning realizzi pienamente la traiettoria aperta dalla cibernetica: una transizione dal calcolo seriale al calcolo distribuito, da un'intelligenza programmata a un'intelligenza che emerge dalle relazioni tra informazione, adattamento e ambiente, compiendo - almeno in parte - la promessa originaria della simbiosi uomo-macchina.

L'analisi condotta a diversi livelli di astrazione, facendo della metafora cibernetica il proprio strumento di indagine, non poteva che ricondurre alle criticità dei modelli contemporanei di intelligenza artificiale, conferendo loro una consistenza insieme storica, concettuale e tecnico-scientifica. Alla luce della metafora analitica impiegata, tali criticità appaiono come lo "sbocciare" del seme originario piantato con il predittore antiaereo, da cui la cibernetica ha avuto origine e che continua - in forme mutate - a orientare la logica della previsione e del controllo nelle architetture dell'IA.

Come l'analisi condotta in questo lavoro ha mostrato le attuali tecnologie connessioniste mostrano limiti strutturali che affondano le proprie radici proprio in questa logica statistico-predittiva. Il deep learning è un paradigma affamato di dati, la cui efficacia cresce con la scala ma decresce con la scarsità informativa. Questi modelli eccellono nell'individuare correlazioni, ma faticano a generalizzare o a inferire oltre i dati forniti, mancando di integrazione con conoscenze pregresse e di una rappresentazione causale. Ne derivano risposte approssimative e opache, incapaci di distinguere il nesso tra causa ed effetto: potenti nell'associare, deboli nel comprendere [Marcus, 2018].

Tali criticità risultano, nella prospettiva qui adottata, "disinnescabili" solo a partire da una conoscenza storica e filosofica delle direttrici che hanno guidato lo sviluppo di questi modelli. Come osserva Ed Finn, gli algoritmi vanno considerati come vere e proprie macchine culturali: complessi aggregati di astrazioni, processi e persone che affondano le proprie radici non solo nella logica matematica, ma anche nelle tradizioni filosofiche della cibernetica, della coscienza e perfino della "magia del linguaggio simbolico" [Finn, 2017, p. viii, trad. it.]. Comprendere tali dispositivi, aggiunge l'autore, richiede pertanto "i metodi critici delle discipline umanistiche" [Finn, 2017, p. viii, trad. it.]. Solo così è possibile comprendere anche lo

shift culturale che dal “mondo chiuso” del digitale [Edwards, 1996] conduce alle architetture connessioniste contemporanee: un passaggio che riflette la crescente interconnettività globale, la logica della planetarizzazione e la prevalenza di dinamiche reticolari basate su interazioni locali. Questi modelli rispecchiano la mutata condizione politico-culturale del presente, segnata dal declino delle forme centralizzate di controllo e dall’emergere di sistemi adattivi, plurali e decentrati.

Ma in quest’ottica l’intelligenza artificiale contemporanea non è esente da rischi, come l’analisi ha mostrato. A questi si aggiungono, nei modelli linguistici di nuova generazione, criticità di natura sociale, ambientale e cognitiva. Tali modelli tendono a riprodurre e amplificare i *bias* presenti nei dati di addestramento, richiedono enormi risorse energetiche e generano l’illusione di una comprensione che in realtà non possiedono. Il loro linguaggio, statisticamente plausibile ma semanticamente vuoto, induce l’utente a proiettare significato dove non vi è alcuna intenzionalità né rappresentazione del mondo [Bender et al., 2021]. Il rischio di delegare prerogative propriamente umane a “*pappagalli stocastici*” rimane dunque costante - e sempre più pressante.

Come evitarlo? L’analisi qui svolta ha proposto la soluzione di considerare le anime dell’IA e delle tecnologie digitali come una forma di *scissione* tra intelligenza e comportamento intelligente. Alla luce di questo divorzio le *Information and Communication Technologies* vanno – alla luce del loro successo ingegneristico strepitoso e del simultaneo fallimento come riproduttori cognitivi di intelligenza- come “*riserva di capacità di agire a portata di mano*.” [Floridi, 2022, p.53] che via via tende ad avvolgere gli ambienti – ne è esempio *l’Internet of Things* [Li et al., 2015] - ad integrarsi nell’architettura cognitiva estesa dell’essere umano [Clark, 2025].

In questo passaggio dall’intelligenza all’azione, le prospettive di ritrovare l’umano dietro i flussi di 0 e 1 non potevano, in questo lavoro, non passare da una presa in esame della dimensione etica del digitale. L’analisi ha evidenziato la mutata fenomenologia etica frutto delle nuove tecnologie [van den Hoven, 2017], segnata da un nuovo quadro di *stakeholder* accomunati dal punto di vista ontologico minimale dell’informazione [Floridi, 2008; Friedman, Hendry, 2019, p. 37], da inedite forme di *accountability* per macchine sempre più autonome [Flusser, 2003] e dal ruolo della mediazione tecnologica [Verbeek, 2006]. Tali elementi convergono nel mostrare la necessità di un’etica progettuale capace di orientare il design verso fini umani e sostenibili, restituendo al soggetto un ruolo attivo nel definire il senso dell’agire

tecnico: verso un'innovazione sostenibile e un'intelligenza artificiale per il bene sociale [AI4SG; Umbrello, 2023; Floridi, 2022].

Il *design* diviene allora il luogo in cui ricomporre il divorzio tra cognizione e azione, tracciando un volto umano nel *blurring* tra intelligenza e capacità computazionale.

Il lavoro qui proposto ha tentato di dare una risposta concreta e applicabile nei luoghi in cui le tecnologie digitali vengono effettivamente progettate e implementate, attraverso il *Framework etico per la progettazione di tecnologie AI-HCI* realizzato durante il periodo di lavoro aziendale di questo progetto di tesi nel team di *design thinking* della DFXM presso Exprivia Spa.

Gli sviluppi delineati nel lavoro- progettato per essere uno strumento immediatamente fruibile per i *designer* - integrano i risultati teorici e critici dell'analisi in una prospettiva operativa, in cui il design diventa pratica riflessiva e spazio di mediazione tra valori, utenti e sistemi intelligenti. Il modello proposto, articolato in sezioni dedicate a *valori etici e principi di progetto, processi decisionali trasparenti, valutazione degli impatti socio-cognitivi e strumenti per il design responsabile*, traduce le istanze dell'etica informazionale in procedure applicabili al ciclo di vita delle tecnologie digitali.

Esempi concreti riguardano la definizione di criteri di *accountability* per l'interazione uomo-macchina, la progettazione di interfacce "simmetriche" capaci di ridurre l'asimmetria informativa tra utente e sistema, o la costruzione di matrici valoriali per valutare le conseguenze etiche delle scelte algoritmiche in fase di sviluppo.

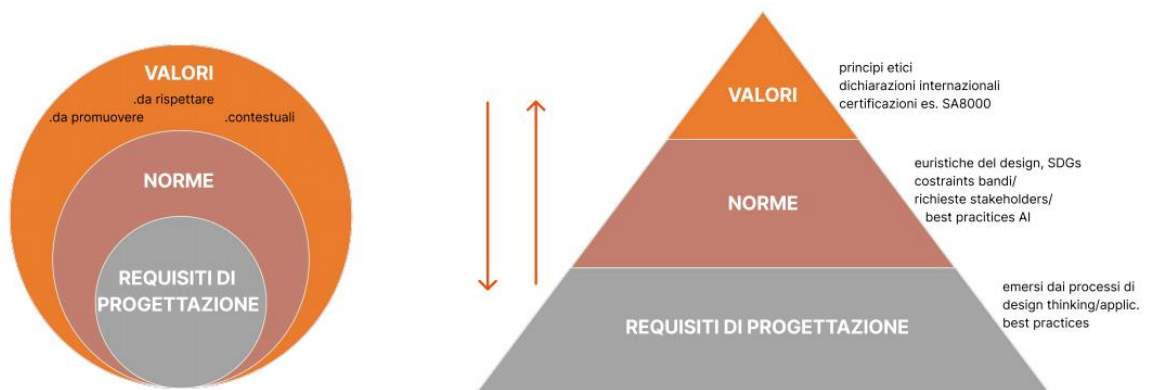
In questo contesto, la figura dell'*eticista designer* [van Wynsberghe, Robbins, 2013; Shilton, Anderson, 2016, Van de Poel, 2013] delineata nel *Framework*, assume un ruolo cruciale come *value advocate* all'interno dei team di progettazione. Essa unisce competenze etiche, epistemologiche e tecniche per chiarire i valori in gioco, mediare tra prospettive diverse e tradurre opzioni normative in scelte progettuali verificabili.

L'eticista non agisce come esperto esterno o censore, ma come facilitatore: un membro stabile del gruppo di lavoro che rende l'etica un *driver* interno del processo di innovazione, evitando derive di "ethical washing" e favorendo un dialogo continuo tra ingegneri, designer e stakeholder.

Operativamente, il suo contributo si esplica nelle attività di identificazione e coinvolgimento degli *stakeholder*, analisi della conoscenza tacita ed esplicita, gestione dei *trade-off* valoriali, mediazione delle tensioni etiche e traduzione dei valori in requisiti di progettazione concreti.

Tale figura incarna la svolta pragmatica dell'etica contemporanea: un'etica situata, iterativa e co-progettuale, capace di tradurre i valori in *affordance* progettuali concrete e di trasformare principi astratti - come autonomia, giustizia, beneficenza, non maleficenza ed esplicabilità - in metriche di *accountability* e in proprietà effettive degli artefatti digitali [Fig. 32]

Fig. 32 Dai valori alle affordances



Il design così può diventare uno strumento che, riverberando verso i valori la tensione sempre presente nella metafora cibernetica da cui è scaturito, per condurre dalla *paura* (veicolata anche dal discorso pubblico) alla *speranza*: la rivoluzione digitale impone di trasformare le tecnologie da strumenti di alienazione a strumenti di emancipazione. Un design etico e consapevole, integrato con politiche adeguate, può orientare l'innovazione verso la sostenibilità, liberando le capacità umane più alte e restituendo senso e valore all'agire tecnico.

Nell'ecosistema di informazioni che caratterizza la contemporaneità digitale, l'impiego di strumenti per ridurre l'*entropia* informativa [Floridi, 2009; Bynum, 2009] – come quelli tentati in questo lavoro - può valorizzare i *sé umani*, intesi come:

le tecnologie *negentropiche* ultime, attraverso cui l'informazione temporaneamente supera la propria entropia, diventa cosciente, e capace di raccontare la storia della propria emergenza in termini di un progressivo distacco dalla realtà esterna. Ci sono ancora solo strutture informazionali. Ma alcune sono cose, alcune sono organismi, e alcune sono menti, esseri intelligenti e auto-consapevoli. Solo le

menti sono capaci di interpretare altre strutture informazionali come cose o organismi o sé. E questa è parte della loro posizione speciale nell'universo [Floridi, 2011b trad mia]².

Per individuare e preservare questa peculiarità dell'essere umano all'interno dell'ecosistema informazionale, il design si configura come lo spazio privilegiato in cui tradurre in pratica la prospettiva critica elaborata nel percorso di ricerca. L'analisi storico-scientifica, tecnologica, formale e critico-filosofica sviluppata in questa indagine ha ricostruito le linee di continuità tra cibernetica, teoria dell'informazione e computazione, mostrando come il loro sviluppo non rappresenti un destino tecnologicamente – in modo tutt'altro che “neutrale” – predeterminata, ma una possibilità ulteriore per la capacità umana di azione e di autodeterminazione. In tale prospettiva, la metafora cibernetica qui impiegata ha operato come principio organizzatore, permettendo di riconoscere nel design non solo un ambito tecnico, ma una forma di riflessione operativa sul nesso tra informazione e valore, tra calcolo e significato. È in questa direzione che la storia delle macchine torna a coincidere con la storia dell'uomo, e la progettazione tecnologica diventa lo strumento per rendere sostenibile, etica e consapevole la nostra presenza nell'età dell'informazione digitale. Il design allora si configura come uno strumento per salvaguardare il ruolo speciale dell'essere umano, considerato anch'esso come struttura di informazioni messa in comunicazione con altri sistemi – artificiali – informazionali, realizzando la promessa emancipativa contenuta sin nella sua origine nella metafora cibernetica: quella di un *uso umano degli esseri umani* [Wiener, 1950].

Le prospettive future di ricerca di questo lavoro, nella continuità del punto di vista analitico adottato, si orientano verso l'analisi delle forme emergenti di interazione uomo-macchina, sempre più tecnologicamente avanzate - dalle interfacce neurali ai sistemi immersivi della realtà virtuale e aumentata - in cui il blurring tra naturale e artificiale raggiunge nuovi livelli di integrazione. Un secondo asse di ricerca riguarda l'approfondimento critico delle modalità di antropomorfizzazione delle macchine e delle loro funzioni, in relazione ai diversi contesti socio-culturali che ne orientano la percezione e l'uso. Parallelamente, l'indagine sulle conseguenze che tali interazioni producono sull'identità umana, chiamata a ridefinirsi nel rapporto con un ambiente digitale sempre più pervasivo, e lo sviluppo dell'analisi storico-scientifica dei vettori dell'HCI potranno fornire una base per orientare criticamente la lettura

² “Selves are the ultimate negentropic technologies, through which information temporarily overcomes its own entropy, becomes conscious, and is finally able to recount the story of its own emergence in terms of a progressive detachment from external reality. There are still only informational structures. But some are things, some are organisms, and some are minds, intelligent and self-aware beings. Only minds are able to interpret other informational structures as things or organisms or selves. And this is part of their special position in the universe”

degli sviluppi contemporanei e contribuire all'elaborazione di protocolli etici integrati, capaci di coniugare competenze tecniche e istanze umanistiche - considerando la privacy, l'autonomia e la rappresentazione del sé come dimensioni costitutive dell'identità personale nell'era informazionale..

BIBLIOGRAFIA

ABBATE, J.

2000 *Inventing the Internet*. Cambridge (MA), MIT Press.

ABRAHAM, T.H.

2002 *(Physio) logical circuits: The intellectual origins of the McCulloch–Pitts neural networks*. *Journal of the History of the Behavioral Sciences*, 38(1):3–25.

ADRIAANS, P.

2020 *Information*, in *Stanford Encyclopedia of Philosophy* (2012).

AHO, A. V.; LAM, M. S.; SETHI, R.; ULLMAN, J. D.

2006 *Compilers: Principles, Techniques, and Tools* (2^a ed.). Addison-Wesley Longman Publishing Co., Inc., Boston (MA).

AIKEN, H.H.

1937 *Proposed Automatic Calculating Machine*, in Randell, B. *The Origins of Digital Computers Selected Papers*, 2 ed. Springer-Verlag Berlin Heidelberg New York 1975, pp.191-198.

AIKEN, H.H., HOPPER G.M.

1946 *The Automatic Sequence Controlled Calculator*, Randell, B. *The Origins of Digital Computers Selected Papers*, 2 ed. Springer-Verlag Berlin Heidelberg New York 1975, pp.199-218.

AIR DEFENSE SYSTEM (ADSEC)

1950 *Final Report*, 24 ottobre 1950, in Turner, F., *From Counterculture to Cyberculture: Stewart Brand, the Whole Earth Network, and the Rise of Digital Utopianism*. University of Chicago Press, 2006.

ALBRECHTSLUND, A.

2007 *Ethics and Technology Design, Ethics and Information Technology*, 9, pp. 63–72.

<https://doi.org/10.1007/s10676-006-9129-8>

ALLEN, C., VARNER, G., ZINSER, J.

2000 *Prolegomena to Any Future Artificial Moral Agent*, *Journal of Experimental & Theoretical Artificial Intelligence*, 12, pp. 251–261.

AMARI, S.

1967 *A Theory of Adaptive Pattern Classifiers*, *IEEE Transactions on Electronic Computers*, EC-16(3), pp. 299–307. <https://doi.org/10.1109/PGEC.1967.264666>

AMDAHL, G. M.; BLAUW, A. G.; BROOKS, F. P. Jr.

1964 *Architecture of the IBM System/360*. *IBM Journal of Research and Development*, 8(2), pp. 87–101.

ANDERSON, J. A.

1972 *A Simple Neural Network Generating an Interactive Memory*, *Mathematical Biosciences*, 14(3–4), pp. 197–220. [https://doi.org/10.1016/0025-5564\(72\)90075-2](https://doi.org/10.1016/0025-5564(72)90075-2)

ANDERSON, J. A., ROSENFELD, E. (eds.)

1988 *Neurocomputing: Foundations of Research*, MIT Press, Cambridge (MA).

ANDERSON, J. A., ROSENFELD, E.

1988 *Introduction to Rochester et al. 1956*, in Anderson, J. A., Rosenfeld, E. (a cura di), *Neurocomputing: Foundations of Research*, MIT Press, Cambridge (MA), pp. 65–67.

1988b *Introduction to Widrow, B., Hoff, M. E. 1960*, in Anderson, J. A., Rosenfeld, E. (a cura di), *Neurocomputing: Foundations of Research*, MIT Press, Cambridge (MA), pp. 123–125.

ARBIB, M. A.

1966 *Simple Self-Reproducing Universal Automata*, *Information and Control*, 9, pp. 177–189.

- 1972 *The Metaphorical Brain: An Introduction to Cybernetics as Artificial Intelligence and Brain Theory*. 1972. Wiley-Interscience, New York.
- 1995 *The Handbook of Brain Theory and Neural Networks*, 1^a ed., MIT Press, Cambridge (MA).
- 2000 *Warren McCulloch's search for the logic of the nervous system*. *Perspectives in biology and medicine*, 43(2):193–216.
- 2002 (a cura di.) Arbib, M. *The Handbook of Brain Theory and Neural Networks*, 2^a ed., MIT Press, Cambridge (MA). ISBN (electronic): 9780262267267. <https://doi.org/10.7551/mitpress/3413.001.0001>
- 2002 *Backpropagation: General Principles*, in *The Handbook of Brain Theory and Neural Networks*, 2^a ed., MIT Press, Cambridge (MA), pp. 144–147.
- 2003 *Backpropagation: General Principles*, in Arbib, M. A. (ed.), *The Handbook of Brain Theory and Neural Networks*, 3^a ed., MIT Press, Cambridge (MA), pp. 144–147.
- ASHBY, W.R.
1940 *Adaptiveness and equilibrium*. In: *J. Ment. Sci.* 86, 478.
- 1951 *Lettera a N. Wiener*, 6 febbraio, in *The Ontology of the Enemy: Norbert Wiener and the Cybernetic Vision*, *Critical Inquiry* 21 (1):228–266 (1994) 10.1086/448747, in (a cura di) Bianchini, F., Franchi, S., Matteuzzi, M., *Verso un'archeologia dell'intelligenza artificiale*, In "Discipline Filosofiche", XVII I, Quodlibet, Macerata, 2007, pp.41–84.
- 1952 *Design for a Brain*, Wiley; 2 ed. 1960 *Design for a brain: The origin of adaptive behaviour* (2nd ed. rev.). Chapman & Hall; trad. it. *Progetto per un cervello*, Bompiani, Milano 1970.
- 1952b *Homeostasis*, in *Cybernetics: The Macy Conferences 1946–1953. The Complete Transactions*, a cura di Claus Pias, Diaphanes, , 2016, pp. 593–560.
- ASPRAY, W.
1990 *John von Neumann and the Origins of Modern Computing*, MIT Press, Cambridge
- ASTRAHAN, M. M.; JACOBS, J. F.
1983 *History of the Design of the SAGE Computer – The AN/FSQ-7*. *IEEE Annals of the History of Computing*, 5(4), pp. 340–349.
- BABBAGE, C.
1826 *On a Method of Expressing by Signs the Action of Machinery*, in «Philosophical Transactions of the Royal Society of London», vol. 116, pp. 250–265.
- 1864 *Passages from the Life of a Philosopher*, Longman, Green, Longman, Roberts, & Green.
- BACKUS, J.
1978 *Can programming be liberated from the von Neumann style? A functional style and its algebra of programs*. *Comm. ACM* 21: 8, 613–641 (1978).
- 1980 *Programming in America in the 1950s — Some Personal Impressions*, in Metropolis et al. (a cura di), *A History of Computing in the Twentieth Century*. Academic Press, New York–London, pp. 125–137.
- BANKS, E. R.
1971 *Information Processing and Transmission in Cellular Automata*, Doctoral dissertation, Vanderbilt University, Nashville (TN).
- BARALE, A.
2020 *Arte e Intelligenza Artificiale. Be My Gan*, Jaca Book, Milano

BARRICELLI, N. A.

1962 *Numerical Testing of Evolution Theories*. *Acta Biotheoretica*, 16, pp. 69–98.

<https://doi.org/10.1007/BF01556771>.

BARDINI, T.

2000 *Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing*, Stanford University Press.

BARBER, G.

1956 *Ballisticians in War and Peace. A History of the United States Army Ballistic Research Laboratories. Volume I: 1914–1956*. Aberdeen Proving Ground, Maryland.

BARLOW, J. P.

1996 *A Declaration of the Independence of Cyberspace*. Davos: Electronic Frontier Foundation.

BARROW, H.

1996 *Connectionism and Neural Networks*, in BODEN, M. A. (a cura di), *Handbook of Perception and Cognition. Artificial Intelligence*, Academic Press, pp. 135–155. <https://doi.org/10.1016/B978-012161964-0/50007-8>

BASHE, C. J.; JOHNSON, L. R.; PALMER, J. H.; PUGH, E. W.

1986 *IBM's Early Computers*. 2nd ed. Cambridge (MA), The MIT Press.

BASHE, C. J.; BUCHHOLZ, W.; HAWKINS, G. V.; INGRAM, J. J.; ROCHESTER, N.

1981 *The Architecture of IBM's Early Computers*. *IBM Journal of Research and Development*, 25(5), pp. 363–375.

BATESON, G.

1955 *A Theory of Play and Fantasy: A Report on Theoretical Aspects of the Project for Study of the Role of Paradoxes of Abstractions in Communication*, in *Approaches to the Study of Human Personality* (American Psychiatric Association. *Psychiatric Reports*, n. 2), pp. 39–51; trad. it. *Una teoria del gioco e della fantasia*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 1992, pp. 217–235.

1956 *Toward a Theory of Schizophrenia*, in *Behavioral Science*, 1 (4), ottobre, pp. 251–264; trad. it. *Verso una teoria della schizofrenia*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 1992, pp. 244–274.

1967 *Cybernetic Explanation*, in *American Behavioral Scientist*, 10 (8), aprile, pp. 29–32; trad. it. *La spiegazione cibernetica*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 2000.

1970 *Form, Substance, and Difference*, in *General Semantics Bulletin*, 37, pp. 5–13. Conferenza per il Diciannovesimo Annual Alfred Korzybski Memorial, tenutasi a New York il 9 gennaio 1970; trad. it. *Forma, sostanza e differenza*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 2000, pp. 464–484.

1971 *The Cybernetics of "Self": A Theory of Alcoholism*, in *Psychiatry*, 34 (1), febbraio, pp. 1–18; trad. it. *La cibernetica dell'io: una teoria dell'alcolismo*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 1992, pp. 339–375.

1972a *The Logical Categories of Learning and Communication*, in *Steps to an Ecology of Mind*; trad. it. *Le categorie logiche dell'apprendimento e della comunicazione*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 1992, pp. 303–338.

1972b *Pathologies of Epistemology*, in *Steps to an Ecology of Mind*; trad. it. *Patologie dell'epistemologia*, in Bateson, G., *Verso un'ecologia della mente*, Adelphi, Milano 1992, pp. 497–508.

BAUER, F. L.

1980 *Between Zuse and Rutishauser—The Early Development of Digital Computing in Central Europe*, in ed. Metropolis, et al. *A History of Computing in the Twentieth Century*, Academic Press, New York.

BECHTEL, W., ABRAHAMSEN, A.

1992 *Connectionism and the Mind: Parallel Processing, Dynamics, and Evolution in Networks*, 1^a ed., Blackwell,

Oxford.

2002 *Connectionism and the Mind: Parallel Processing, Dynamics, and Evolution in Networks*, 2^a ed., John Wiley & Sons, Hoboken

BEDAU, M. A.

2003 *Artificial Life: Organization, Adaptation and Complexity from the Bottom Up*. *Trends in Cognitive Sciences*, 7(11), pp. 505–512.

BERTO, F.; TAGLIABUE, J.

2023 *Cellular Automata*, in E. N. Zalta (ed.), *The Stanford Encyclopedia of Philosophy* (Fall 2023 Edition).

BENDER, E. M., GEBRU, T., McMILLAN-MAJOR, A., SHMITCHELL, S.

2021 *On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?*, in «Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency», pp. 610-623, Association for Computing Machinery. <https://doi.org/10.1145/3442188.3445922>

BERANEK, L.

2000 *Roots of the Internet: A Personal History*. *Massachusetts Historical Review*, 2, pp. 55–75. Boston, Massachusetts Historical Society.

BERDICHEVSKY, D., AND NEUENSCHWANDER, E.

1999 *Toward an ethics of persuasive technology*. *Communications of the ACM* 42, 5, pp.51--58.

BERKELEY, E.C.

1949 *Giant brains; or, Machines that think*. Wiley, New York.

BERNARD, C.

1848 *Du suc pancreatique et de son role dans les phenomenes de la digestion*. *Memoires du Societe´ de Biologie*, 1, 99–115.

BERNERS-LEE, T.

1999 *Weaving the Web. The Original Design and Ultimate Destiny of the World Wide Web*. 1st ed. New York, HarperCollins Publishers.

BEYER, K. W.

2009 *Grace Hopper and the Invention of the Information Age*, MIT Press, Cambridge (MA)

BHATTACHARYA, A.

2021 *The Man from the Future: The Visionary Life of John von Neumann*. Penguin Books, London.

BIANCHINI, F.

2007 *Le trasformazioni del Test di Turing da Cartesio a Leibniz*, in Bianchini et. al. (a cura di) *Verso un'archeologia dell'intelligenza artificiale*, in *Discipline Filosofiche* I n.17, Quodlibet, Macerata.

BIEVER, C.

2023 *Chatgpt broke the turing test-the race is on for new ways to assess ai*, in *Nature*, 619(7971):686–689.

BIGELOW, J.

1980 *Computer Development at the Institute for Advanced Study*, in in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.291-310.

BLOCK, N:

1980 *What is Functionalism?*, in N. Block (a cura di), *Readings in Philosophy of Psychology: Volume 1*, Cambridge (MA): Harvard University Press.

BODEN, M.

1996 *The Philosophy of Artificial Life*. Oxford University Press, Oxford.

- 1996 *Autonomy and Artificiality*, in M. A. Boden (ed.), *The Philosophy of Artificial Life*. Oxford University Press, Oxford.
- 2006 *Mind as Machine: A History of Cognitive Science*, Oxford University Press.
- 2018 *Artificial Intelligence: A Very Short Introduction*, Oxford University Press., New York; trad. it. *Intelligenza artificiale* Il Mulino, Bologna 2021.
- BONFANTI, M.
 2004 *L'alba dell'informatica in Italia: il CRC 102A e il FINAC*, in *Storia e Futuro*, n. 5, marzo 2004. Disponibile online: storiaefuturo.eu (consultato 2025).
- BOOLE, G.
 1847 *The Mathematical Analysis of Logic, Being an Essay towards a Calculus of Deductive Reasoning*, Macmillan, Barclay, & Macmillan, Cambridge and George Bell, Londra
- 1854 *An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities*, Walton and Maberly, Londra.
- BOSTROM, N.
 2002 *Existential Risks: Analyzing Human Extinction Scenarios and Related Hazards*, *Journal of Evolution and Technology*, 9(1).
- 2003 *Ethical Issues in Advanced Artificial Intelligence*, in *Cognitive, Emotive and Ethical Aspects of Decision Making in Humans and in Artificial Intelligence*, Vol. 2.
- 2014 *Superintelligence. Paths, Dangers, Strategies*, Oxford University Press, Oxford; trad. it. *Superintelligenza. Tendenze, pericoli, strategie*, Bollati Boringhieri, Torino, 2018.
- BORNE, K.
 2014 *Top 10 Big Data Challenges — A Serious Look at 10 Big Data V's*. *Data Science Central*, 11 aprile.
- BOULDING, K. E.
 1970 *Economics as a Science*. McGraw-Hill.
- BRAND, S.
 1972 *Spacewar: Fanatic Life and Symbolic Death Among the Computer Bums*, *Rolling Stone*, December 7.
- BRAND, S (ed.)
 1968 *Whole Earth Catalog*, Fall Issee.
- BRAUTIGAN, R.
 1967 *All Watched Over by Machines of Loving Grace*, Communication Company, San Francisco.
- BRINGSJORD, S., GOVINDARAJULU, N. S.
 2018 *Artificial Intelligence*, in *Stanford Encyclopedia of Philosophy*.
- BRINKMAN, W. F.; HAGGAN, D. E.; TROUTMAN, W. W.
 1997 *A History of the Invention of the Transistor and Where It Will Lead Us*. *IEEE Journal of Solid-State Circuits*, 32(12), pp. 1858–1865.
- BROMLEY, A. G.
 1987 *Charles Babbage's Analytical Engine, 1838*, *IEEE Annals of the History of Computing*, Vol. 9, No. 2, pp. 113-136.
- BROOKS, F. P. Jr.
 1975 *The Mythical Man-Month. Essays on Software Engineering*. Addison-Wesley Publishing Company.

- 1986 *No Silver Bullet – Essence and Accidents of Software Engineering. Proceedings of the IFIP Tenth World Computing Conference*, pp. 1069–1076; *IEEE Computer*, 20(4), 1987, pp. 10–19.
<https://doi.org/10.1109/MC.1987.1663532>.
- 2010 *The Design of Design. Essays from a Computer Scientist*. Addison-Wesley Publishing Company.
- BROOKS, R. A.
 1991 *Intelligence Without Representation, Artificial Intelligence*, 47, pp. 139–159.
- BRUCE, J.; NG, S. W.; WANG, D. T.
 2008 *Memory Systems: Cache, DRAM, Disk*. Con contributi di S. Rodriguez. Burlington (MA), Morgan Kaufmann / Elsevier.
- BRUDERER, H.
 2020 *Milestones in Analog and Digital Computing* (3rd ed.). Springer, Cham.
- BUCHHOLZ, W.
 1953 *The System Design of the IBM Type 701 Computer. Proceedings of the IRE*, 41(10), pp. 1262–1275. doi: 10.1109/JRPROC.1953.274300.
- BUCKNER, C.
 2019 *Deep Learning: Philosophical Introduction, Philosophy Compass*.
- BUCKNER C., GARSON J.
 2019 *Connectionism and post-connectionist models in M. Spreveak, M.Colombo, The Routledge Handbook Of The Computational Mind*, Routledge, Londra, pp.76-90.
- BUDIANSKY, S.
 2006 *Colossus, Codebreaking and the Digital Age*, in ed. Copeland, B. J, and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006 pp.52-63
- BURKE, C.
 1994 *Information and Secrecy: Vannevar Bush, Ultra, and the Other Memex*. Metuchen (NJ), Scarecrow Press.
- 2007 *History of Information Science*. In Cronin, B. (Ed.), *Annual Review of Information Science and Technology*, 41, pp. 3–53. Medford, NJ: Information Today/ASIST. DOI:10.1002/aris.2007.1440410108.
- 2002 *It Wasn't All Magic: The Early Struggle to Automate Cryptanalysis, 1930s–1980s* Center For Cryptologic History, National Security Agency, declassified, 29/05/2013.
- BURKS, A.W.
 1980 *From ENIAC to the Stored-Program Computer: Two Revolutions in Computers*, in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.311-344.
- BURKS, A.R., BURKS, A. W.
 1988 *The First Electronic Computer: The Atanasoff Story*. Ann Arbor, Michigan: The University of Michigan Press. ISBN 978-0-472-10090-3.
- BURKS, A.W., GOLDSTINE, H. H., VON NEUMANN, J.
 1946 *Preliminary discussion of the logical design of an electronic computing instrument*, in Perspectives on the computer revolution, June 1989
- (ed.) BURKS, A. W.; VON NEUMANN, J.
 1966 *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign.
- BUSH, V.
 1929 *Operational Circuit Analysis*. John Wiley & Sons, New York.
- 1931 *The differential analyzer. A new machine for solving differential equations*. MIT.
[https://doi.org/10.1016/S0016-0032\(31\)90616-9](https://doi.org/10.1016/S0016-0032(31)90616-9)

1945 *As We May Think*. In *The Atlantic Monthly*, July.

BUTTERWORTH, B.

1999 *What Counts - How Every Brain is Hardwired for Math*. The Free Press, New York. 2005 The development of arithmetical abilities. *Journal of Child Psychology and Psychiatry*, 46, 3-18.

BYNUM, T. W.

2009 *Filosofia e rivoluzione dell'informazione*, in Floridi, L., *Infosfera. Etica e filosofia nell'età dell'informazione*, Giappichelli, Torino, pp. 3-23.

2010 *Philosophy in the Information Age*, *Metaphilosophy*, 41(3), April, pp. 420-442.

(ed.) BYNUM, T. W.; MOOR, J. H.

1998 *The Digital Phoenix. How Computing Are Changing Philosophy*. Blackwell.

BYNUM, T. W.; MOOR, J. H.

1998 *How Computers Are Changing Philosophy*, in (ed.) Bynum, T. W., Moor, J. H., *The Digital Phoenix. How Computing Are Changing Philosophy*, Blackwell, pp. 1-14.

CAMPBELL-KELLY, M.

2017 *ACE*, in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.183-188.

CAMPBELL-KELLY, M.; WILLIAMS, M. R.

1985 *Introduction*, in *The Moore School Lectures: Theory and Techniques for Design of Electronic Digital Computers*, a cura di M. Campbell-Kelly e M. R. Williams. MIT Press, Cambridge (MA), pp. xiii-xxiv.

CANTOR, G.

1874 *Ueber eine Eigenschaft des Inbegriffs aller reellen algebraischen Zahlen*, in *Journal für die reine und angewandte Mathematik*, 77, pp. 258-262.

CAPURRO, R., HJØRLAND, B.

2003 *The Concept of Information*, *Annual Review of Information Science and Technology*, 37, pp. 343-411.

CARBONE, B.

1979 *Parliamo un po' del 6502. Bit*, Anno 2, No. 5, Milano, Gruppo Editoriale Jackson, pp. 35-40.

CARD, S. K., MORAN, T. P., NEWELL, A.

1983 *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale (NJ).

CARPENTER B., DORAN, R

2017 *Turings's Zeitgeist* in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.223-231.

CARTUYVELS, R., SPINKS, G., MOENS, M.-F.

2021 *Discrete and continuous representations and processing in deep learning: Looking forward*. In *AI Open*, 2, 143-159.

CASTELLUCCI, P.

2009 *Dall'ipertesto al Web. Storia culturale dell'informatica*. Roma-Bari, Laterza.

CAVELL, S.,

1988 *The Uncanniness of the Ordinary*, in *In Quest of the Ordinary :Lines of Skepticism and Belief*, University of Chicago Press, Chicago, pp.153-178.

CERUZZI, P. E.

2003 *A History of Modern Computing* (2^a ed.), collana *History of Computing*, MIT Press, Cambridge (MA).

2012 *Computing: A Concise History*, The MIT Press.

- CHIAVERINI, J.
2017 *Enchantress of Numbers: A Novel of Ada Lovelace*, Dutton.
- CHALMERS, D. J.
2022 *Reality+: Virtual Worlds and the Problems of Philosophy*, W. W. Norton & Company, New York.
- CHOMSKY, N.
1956 *Three Models for the Description of Language*, *IRE Transactions on Information Theory*, 2(3), pp. 113–124.
<https://doi.org/10.1109/TIT.1956.1056813>
- CHURCH, A.
1936 *An Unsolvable Problem of Elementary Number Theory*, *American Journal of Mathematics*, vol. 58, pp. 345–363.
- CHURCHLAND, P., CHURCHLAND, P. S.
2000 *Preface to the second edition*, in von Neumann, J., *The Computer and the Brain*, 2000 (2^a ed.); trad. it. *Prefazione alla seconda edizione*, in von Neumann, J., *Computer e Cervello*, Il Saggiatore, Milano 2021, pp. 31–42.
- CHURCHLAND, P., SEJNOWSKI, T. J.
1990 *Neural Representation and Neural Computation* in *Philosophical Perspectives*, Vol. 4, Action Theory and Philosophy of Mind (1990), pp. 343–382.
- CIOTTI, F.
2023 *Digital Humanities: metodi, strumenti, saperi*. Il Mulino, Bologna.
(a cura di) CIOTTI, F.
2014 *Digital Humanities: progetti italiani ed esperienze di convergenza multidisciplinare. Atti del convegno annuale dell'Associazione per l'Informatica Umanistica e la Cultura Digitale (AIUCD)*, Firenze, 13–14 dicembre 2012.
- CLARK, A.
1989 *Microcognition: Philosophy, Cognitive Science, and Parallel Distributed Processing*, MIT Press, Cambridge MA.
2025 *Extending Minds with Generative AI*. In *Nature Communications*, 16, 4627. <https://doi.org/10.1038/s41467-025-59906-9>
- CLARK, W. A.
1986 *The LINC Was Early and Small*. In *Proceedings of the ACM Conference on the History of Personal Workstations*. Palo Alto (CA), ACM, pp. 133–155.
- COHEN, I.B.
2003 *Mark I*, *Harvard*, *Encyclopedia of Computer Science*, January 2003, pp.1078 – 1080.
1999 *Howard Aiken: Portrait of a Computer Pioneer*, MIT Press, Cambridge (MA).
- COLBY, K. M., HILF, F. D., WEBER, S., KRAEMER, H.
1972 *Turing-like indistinguishability tests for the validation of a computer simulation of paranoid processes*, *Artificial Intelligence*, 3: 199–221, doi:10.1016/0004-3702(72)90049-5.
- COUTO, M., PEREIRA, R., RIBEIRO, F., RUA, R., SARAIVA, J.
2017 *Towards a Green Ranking for Programming Languages*, in *SBLP '17: Proceedings of the 21st Brazilian Symposium on Programming Languages*, Article No. 7, pp. 1–8. <https://doi.org/10.1145/3125374.3125382>.
- BERLEKAMP, E. R., CONWAY, J. H., GUY, R. K.
1982 *Winning Ways for Your Mathematical Plays*, 2 vols., Academic Press, New York.
- CONWAY, F., SIEGELMAN, J.
2005 *Dark Hero of the Information Age: In Search of Norbert Wiener, the Father of Cybernetics*, Basic Books; trad. it. *L'eroe oscuro dell'età dell'informazione*, Codice, Torino, 2019.

COOK, M.

2004 *Universality in Elementary Cellular Automata*. *Complex Systems*, 15(1), pp. 1–40.

COPELAND, B. J. (a cura di)

2004 *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*, Oxford University Press, Oxford.

COPELAND, B. J.

2000 *The Turing Test*, in *Minds and Machines*, 10 (4), pp. 519–539.

2004 *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*, Oxford University Press.

2004b *Artificial Intelligence*, in ed. Copeland B. J., *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma* Oxford University Press, Oxford 2004 pp.353-361.

2004cc *Introduction to Intelligent Machinery* in ed. Copeland B. J., *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma* Oxford University Press, Oxford 2004 pp.395-409

2005a *Introduction* in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.1-11.

2005b *The origins and development of the ACE project* in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.47-92.

2006a *Introduction* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.1-8.

2006b *The German Tunny Machine*, in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp. 36-51.

2006c *Intercept!* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp. 84-90.

2006d *Machine against Machina* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.64-77.

2006e *Turingery* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.378-385.

2006f *Colossus and the Rise of the Modern Computer* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.101-115.

2011a *The Manchester Computer: A Revised History. Part 1: The Memory*, in *IEEE Annals of the History of Computing*, 33(1), gennaio 2011, pp. 4–21.

2011b *The Manchester Computer: A Revised History. Part 2: The Baby Computer*, in *IEEE Annals of the History of Computing*, 33(1), febbraio 2011, pp. 22–37.

2012 *Turing: Pioneer of the Information Age*, Oxford University Press, Oxford; trad. it, *Turing, un pioniere dell'età dell'informazione*, Mondadori, Milano, 2017 .

- 2017 *Tunny: Hitler's Biggest Fish* in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.143-160.
- 2017b *Delilah – Encrypting speech*, in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.183-188.
- 2017c *Turing's great invention: the universal computing machine* in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.49-56.
- COPELAND, B.J., CAUGHEY, C., DU BOISSON, D., IRELAND, E.
2006 *Mr Newman's Section* in ed. Copeland, B. J, and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp. 157-175.
- COPELAND, B. J., PRINZ, D.
2017 *Computer-chess. The first moments*, in Copeland, B. J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp. 327-346.
- COPELAND, B. J., PROUDFOOT, D.
1996 *On Alan Turing's anticipation of connectionism*, *Synthese* 108, 361-377
<https://doi.org/10.1007/BF00413694>. 1999 *Alan Turing's Forgotten Ideas in Computer Science*, *Scientific American* 280 (April), pp. 76-81.
- 1999 *Alan Turing's Forgotten Ideas in Computer Science*, *Scientific American*, vol. 280, n. 4, pp. 98-103.
- 2005 *Turing and the Computer*, in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.107-148.
- COPELAND, B. J.; FAN, Z.
2023 *Turing and von Neumann: From Logic to the Computer*, in *Philosophies*, 8 (22).
<https://doi.org/10.3390/philosophies8020022>.
- CORDESCHI, R.
1994 *Indagini meccanicistiche sulla mente: la cibernetica e l'intelligenza artificiale*, in Somenzi, V., Cordeschi, R, *La filosofia degli automi. Origini dell'intelligenza artificiale*, Bollati Boringhieri, Torino 1994, pp. 21-61.
- 1998 *La scoperta dell'artificiale: Origini della cibernetica e dell'intelligenza artificiale*, Dunod.
- CORDESCHI, R., TAMBURRINI, G.
2015 *Alan Turing e il programma di ricerca dell'intelligenza artificiale*, in Hosni H., a cura di, *Menti e macchine. Alan Mathison Turing a cento anni dalla nascita*, Pisa: Edizioni della Normale: 87-126.
- CRAIK, K. J. W.
1943 *The nature of explanation*, Cambridge University Press
- CRAWFORD, K.
2021 *Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*, Yale University Press, New Haven; trad. it. *Né intelligente né artificiale: il lato oscuro dell'IA*, Il Mulino, Bologna 2021 2023.
- CROARKEN, M.
1993 *The Beginnings of the Manchester Computer Phenomenon: People and Influences* in *IEEE Annals of the History of Computing*, Volume 15, Issue 3, pp.9 – 16.
- 2005 *The Creation of NLP Mathematical Division*, in in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.23-36.
- CURBERA, J.
2009 *Mathematicians of the World, Unite!: The International Congress of Mathematicians--A Human Endeavor*, A K Peters/CRC Press.

DATTERI, E.

2017 *Che cos'è la scienza cognitiva*, Carocci Milano.

DAILY TELEGRAPH

1946 *Britain to Make a Radio Brain*, in *Daily Telegraph*, 7 novembre 1946.

1946 *ACE Superior to U. S. Model*, in *Daily Telegraph*, 7 novembre 1946.

1950 *ACE may be the Fastest Brain*, in *Daily Telegraph*, 31 gennaio 1950.

DI FRANCESCO, M.

2002 *Introduzione alla filosofia della mente*, Carocci, Roma.

DAVIS, M.

1958 *Computability and Unsolvability*, New York: McGraw-Hill. 2000.

2000 *The Universal Computer. The road from Leibniz to Turing*, W. W. Norton & Co.; trad. it. *Il calcolatore universale. Da Leibniz a Turing*, Adelphi, Milano, 2000.

2006 *Why there is no such discipline as hypercomputation* in *Applied Mathematics and Computation*, Volume 178, Issue 1.

DAVIS, J., NATHAN, L. P.

2015 *Value Sensitive Design: Applications, Adaptations, and Critiques*, in VAN DEN HOVEN, J., VERMAAS, P., VAN DE POEL, I. (eds.), *Handbook of Ethics, Values, and Technological Design*, Springer, Dordrecht.

https://doi.org/10.1007/978-94-007-6970-0_3

DAWKINS, R.

1976 *The Selfish Gene*. Oxford University Press; trad. it. *Il gene egoista*, Mondadori, Milano 2022.

DAWSON JR., J. W.

1997 *Logical Dilemmas. The Life and Work of Kurt Gödel*, Wellesley, Mass.: A.K. Peters, 1997. Trad. it.: *Dilemmi logici: La vita e l'opera di Kurt Gödel*, traduzione di Paolo Pagli, Torino, Bollati Boringhieri, 2001.

DE CEGLIA, F. P.

2014 *La "grave indecenza" di Alan. Scienza e omosessualità maschile nell'Europa della prima metà del Novecento*, in C. PETROCELLI (a cura di), *Alan Mathison Turing: l'indcidibilità della vita*, Franco Angeli, Milano, pp. 101-116.

2023 *Vampyr. Storia naturale della resurrezione*, Einaudi, Torino.

DEHAENE, S.

2011 – *The Number Sense: How the Mind Creates Mathematics* (Rev. ed.), Oxford University Press, Oxford.

DENNETT, D. C.

1987 *The Intentional Stance*. Cambridge, Mass. MIT Press.

1997 *When Hal Kills, Who's to Blame? Computer Ethics*, in STORK, D. (ed.), *Hal's Legacy: 2001's Computer as Dream and Reality*, MIT Press, Cambridge (MA), pp. 351–365.

DESCARTES, R.

1637 *Discours de la méthode pour bien conduire sa raison, et chercher la verité dans les sciences*, trad. it. *Discorso sul metodo per ben condurre la propria ragione e ricercare la verità nelle scienze*, in *Opere 1637-1649*, Bompiani, Milano, 2009.

1641 *Meditationes de Prima Philosophia*; trad. it. *Meditazioni di filosofia prima* in *Opere 1637-1649*, Bompiani, Milano, 2009.

- DEWEY, M.
1876 *Classification and Subject Index for Cataloguing and Arranging the Books and Pamphlets of a Library*. Hartford, Conn.: Case, Lockwood & Brainard Company.
- DEWHURST, J.
2019 *British Cybernetics* in M. Spreveak, M. Colombo (a cura di), *The Routledge Handbook of the Computational Mind*, Routledge, pp. 38-51.
- DICK, P. K.
1968 *Do Androids Dream of Electric Sheep?*, New York: Doubleday.
- DIEBOLD, F. X.
2012 *On the Origin(s) and Development of the Term "Big Data"* (September 21, 2012). *PIER Working Paper No. 12-037*.
- DIFFIE, W.
2015 *Cryptanalytic Significance of the Analysis of Tunny* in ed. Reeds, J.A, Diffie, W., Field, J.V., *Breaking Teleprinter Ciphers at Bletchley Park*, John Wiley & Sons, Hoboken (NJ) 2015, pp.xvii-xxiv.
- DREYFUS, H.
1972 *What Computers Can't Do: The Limits of Artificial Intelligence*. Harper Colophon Books; trad. it. *Che cosa non possono fare i computer. I limiti dell'intelligenza artificiale*, Armando, Roma, 1988.
- DODIG-CRNKOVIC, G.
2011 *Significance of models of computation, from turing model to natural computation*, *Minds and Machines*, 21, 301-322. doi: 10.1007/s11023-011-9235-1.
- DOMINGOS, P.
2015 *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*, Basic Books; trad. it. *L'Algoritmo definitivo. La macchina che impara da sola e il futuro del nostro mondo*, Bollati Boringhieri, Torino, 2016.
- DORIN, A.; STEPNEY, S.
2024 *What Is Artificial Life Today, and Where Should It Go?* *Artificial Life*, 30(1), pp. 1-15.
<https://doi.org/10.1162/artl.e.00435>.
- DUDLEY, H.
1940 *The Carrier Nature of Speech*, *Bell System Technical Journal*, XIX(4);495-515.
- DYSON, F.
1979 – *Disturbing the Universe*, Basic Books, New York.
- DYSON, G.
2012 *Turing's Cathedral: The Origins of the Digital Universe*, Pantheon Books; trad. it. *La cattedrale di Turing: La genesi dell'universo digitale*, Codice Edizioni, Torino, 2013.
- ECKERT, J. P.
1940 *Punched card methods in scientific computation*. Columbia University, NY.
- 1944 *Disclosure of Magnetic Calculating Machine*, Univ. of Pennsylvania, Moore School of Electrical Eng., memorandum 29 gennaio.
- 1980 *The ENIAC* in in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.525-540.
- 1985 *A Preview of a Digital Computing Machine*, in *The Moore School Lectures: Theory and Techniques for Design of Electronic Digital Computers*, a cura di M. Campbell-Kelly e M. R. Williams. MIT Press, Cambridge (MA), pp. 109-128.

ECKHARDT, R.

1975 *Stan Ulam, John von Neumann, and the Monte Carlo Method*. *Los Alamos Science*, 15, pp. 131–137.

EDMONDS, D., EIDINOW, J.

2001 *Wittgenstein's Poker: The Story of a Ten-Minute Argument Between Two Great Philosophers*, New York: HarperCollins.

2020 *The Murder of Professor Schlick. The rise and the Fall of the Vienna Circle*, Princeton University Press.

EDWARDS, P. N.

1996 *The Closed World. Computers and the Politics of Discourse in Cold War America*. Cambridge (MA), MIT Press.

ELECTRONIC NEWS

1971 *Announcing a New Era of Integrated Electronics. A Micro-Programmable Computer on a Chip!* Advertisement (Intel 4004), November 1971.

ENGELBART, D. C.

1962 *Augmenting Human Intellect: A Conceptual Framework*. Summary Report AFOSR-3223 under Contract AF 49(638)-1024, SRI Project 3578 for Air Force Office of Scientific Research. Menlo Park, Ca., Stanford Research Institute.

EVERETT, R. R.

1989 *WHIRLWIND*, in Metropolis et al. (a cura di), *A History of Computing in the Twentieth Century*. Academic Press, New York–London, pp. 365–384.

EVERETT, R. R.; ZRAKET, C. A.; BENNINGTON, H. D.

1983 *SAGE – A Data Processing System for Air Defense*. *IEEE Annals of the History of Computing*, 5(4), pp. 330–339. <https://doi.org/10.1109/MAHC.1983.10096>.

EVENING NEWS

1946 [Intervista ad Alan Turing], in *Evening News*, 23 dicembre 1946 (cit. in Copeland, B. J.; Proudfoot, D., *Alan Turing: The Legacy*, Oxford University Press, 2005, p. 113).

FAGGIN, F.

2018 *How We Made the Microprocessor*. *Nature Electronics*, 1, p. 88. <https://doi.org/10.1038/s41928-017-0014-8>. 2020 *Silicio. Dall'invenzione del microprocessore alla nuova scienza della consapevolezza*. Milano, Mondadori.

FAGGIN, F.; HOFF, M. E.; MAZOR, S.; SHIMA, M.

1996 *The History of the 4004*. *IEEE Micro*, 16(6), pp. 10–20. doi:10.1109/40.546561.

FANO, R. M.; CORBATÓ, F. J.

1966 *Time-Sharing on Computers*. *Scientific American*, 215(3), pp. 128–143.

FAYOL, M.; SERON, X.

2005 – *About Numerical Representations: Insights from Neuropsychological, Experimental, and Developmental Studies*, in J. I. D. CAMPBELL (a cura di), *Handbook of Mathematical Cognition*, Psychology Press, New York.

FENSOM, H.

2006 *How Colossus was Built and Operated—One of its Engineers Reveals its Secrets* in ed. Copeland, B. J, and Others, *Colossus, The Secrets of Bletchley Par's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.297–306.

FEFERMAN, S.

2006 *Turing's Thesis*, Notices of the AMS, Vol. 53, No. 10.

FEKETE, M.; VON NEUMANN, J.

1922 – *Über die Lage der Nullstellen gewisser Minimumpolynome*, *Mathematische Zeitschrift*.

- FERGUSON, M. T.
2015 *Two Paradoxes of Semantic Information*. *Synthese*, 192(11), pp. 3719–3730.
- FERRY, D. K.; SACKS, R. E.
1987 – *Comments on Wiener's Memorandum*, *Annals of the History of Computing*, 9, pp. 183–197.
- FINN, E.
2017 *What Algorithms Want: Imagination in the Age of Computing*, MIT Press; trad. it. *Che cosa vogliono gli algoritmi. L'immaginazione nell'era dei computer*, Einaudi, Torino, 2020.
- FLORIDI, L.
1999 *Philosophy and Computing: An Introduction*. Routledge.
- 2005 *Open Problems in the Philosophy of Information*, in «Metaphilosophy», vol. 35, n. 4, pp. 554–582. DOI: 10.1111/j.1467-9973.2004.00336.x trad. it. 2005 *La Filosofia dell'informazione e i suoi problemi*, *Iride* 18(45):291–312. DOI:10.1414/20614.
- 2008 *The Method of Levels of Abstraction*, in “Minds and Machines”, 18(3), pp. 303–329.
<https://doi.org/10.1007/s11023-008-9113-7>.
- 2008 *Foundations of Information Ethics*, in HİMBERG, J. M., TAVANI, H. (eds.), *The Handbook of Information and Computer Ethics*, John Wiley & Sons, Hoboken (NJ), pp. 3–23.
- 2009 *L'etica dell'informazione e il fondamento filosofico dell'etica informatica*, in *Infosfera. Etica e filosofia nell'era dell'informazione*, Giappichelli, Torino.
- 2009b *Il male artificiale nella fondazione dell'etica informatica*, in Floridi, L., *Infosfera. Etica e filosofia nell'età dell'informazione*, Giappichelli, Torino, pp. 59–76.
- 2009c *La moralità degli agenti artificiali*, in FLORIDI, L., *Infosfera. Etica e filosofia nell'era dell'informazione*, Giappichelli, Torino, pp. 105–131
- 2010 *Information. A Very Hhort Introduction*; Oxford University Press, trad. it. *La rivoluzione dell'informazione*, Codice, Torino 2012.
- 2011 *The Philosophy of Information*, Oxford University Press, Oxford; trad. it. parz, *La filosofia dell'informazione*, Cortina, Milano, 2024.
- 2011b *The Informational Nature of Personal Identity*, *Minds and Machines*, 21(4), pp. 549–566.
- 2014 *The Fourth Revolution: How the Infosphere is Reshaping Human Reality*, Oxford University Press, Oxford; trad. it. *La quarta rivoluzione: Come l'infosfera sta trasformando il mondo*, Raffaello Cortina, Milano, 2017.
- 2016 *Tolerant Paternalism: Pro-ethical Design as a Resolution of the Dilemma of Toleration*, SSRN, 1 dicembre 2016. <https://ssrn.com/abstract=3835901>
- 2019 *The Logic of information*, Oxford University Press, Oxford; trad. it parz. *Pensare l'infosfera. La filosofia come design concettuale*, Raffaello Cortina, Milano, 2020.
- 2020 *The Fight for Digital Sovereignty: What It Is, and Why It Matters, Especially for the EU*, *Philosophy & Technology*, 33, pp. 369–378. <https://doi.org/10.1007/s13347-020-00423-6>
- 2022 *The Ethics of Artificial Intelligence. Principles, Challenges and Opportunities*, Oxford University Press, Oxford; trad. it. parz. *Etica dell'Intelligenza Artificiale. Sviluppi, opportunità, sfide*, Raffaello Cortina, Milano, 2022.
- 2025 *AI and Semantic Pareidolia: When We See Consciousness Where There Is None*, SSRN, 18 giugno 2025.
<https://ssrn.com/abstract=5309682>
- FLORIDI, L., TADDEO, M., TURILLI, M.

- 2009 *Turing's Imitation Game: Still a Challenge for Any Machine and Some Judges*, *Minds and Machines*, 19, 145–150.
- FLORIDI, L., SANDERS, J.
2004 *On the Morality of Artificial Agents*, *Minds and Machines*, 14, pp. 349–379.
- FLORIDI, L., CHIRIATTI, M.
2020 *GPT-3: Its Nature, Scope, Limits, and Consequences*, *SSRN Electronic Journal*, 1 novembre 2020. Disponibile su: <https://ssrn.com/abstract=3827044> o <http://dx.doi.org/10.2139/ssrn.3827044>.
- FORRESTER, J. W.
1983 *Reliability of Components*. *IEEE Annals of the History of Computing*, 5(4), pp. 399–401.
<https://doi.org/10.1109/MAHC.1983.10099>.
- FLOWERS, T. H.
1983 *The Design of Colossus* (foreword by Howard Campaigne), in *Annals of the History of Computing*, vol. 5, no. 3, pp. 239–252, July–Sept. 1983,
2006 *Colossus*, in in ed. Copeland, B. J. and Others, *Colossus, The Secrets of Bletchley Par's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.91–100.
- FLUSSER, V.
2003 *Etica del design industriale*, in FLUSSER, V., *Filosofia del Design*, Paravia Bruno Mondadori, Milano, pp. 63–67.
- FODOR, J. A., PYLYSHYN, Z. W.
1988 *Connectionism and cognitive architecture: A critical analysis*, *Cognition*, 28(1–2), 3–71.
- FOGG, B. J.
2003 *Persuasive Technology: Using Computers to Change What We Think and Do*, Elsevier; trad. it. *Tecnologia della Persuasione*, Apogeo, Santarcangelo di Romagna, 2017.
- FOURIER, J.
1822 *Théorie analytique de la chaleur*, Firmin-Didot, Paris.
- FRANKEL, S.
1972 Lettera a *Brian Randell*, in RANDELL, B., *On Alan Turing and the Origins of Digital Computers*, in MELTZER, B.; MICHIE, D. (eds.), *Machine Intelligence 7*, Edinburgh University Press, Edinburgh 1972,
- FREGE, G.
1879 *Begriffsschrift. Eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*, Halle: Verlag von Louis Nebert. 1884 *Die Grundlagen der Arithmetik. Eine logisch-mathematische Untersuchung über den Begriff der Zahl*, Breslau: Wilhelm Koebner.
1892 *Über Sinn und Bedeutung*, in *Zeitschrift für Philosophie und philosophische Kritik*, 100, pp. 25–50.
1893–1903 *Grundgesetze der Arithmetik. Begriffsschriftlich abgeleitet*, Jena: Hermann Pohle.
- FREUD, S.
1919 *Das Unheimliche*, trad. it. in. *Leonardo e altri scritti*, Bollati Boringhieri, Torino, 1967, pp. 267–307.
- FRITZ, W.B.
1996 *The Women of ENIAC*. *IEEE Ann. Hist. Comput.* 18, 3 (Fall 1996), 13–28.
- FRIXIONE, M.
1992 *I modelli della mente tra simboli e neuroni: il connessionismo subsimbolico di Smolensky* in P. Smolensky, *Il connessionismo tra simboli e neuroni*, Marietti, Torino, 1992.

FRIEDMAN, B., HENDRY, D. G.

2019 *Value Sensitive Design: Shaping Technology with Moral Imagination*, MIT Press, Cambridge (MA).

FRIEDMAN, B., KAHN, P. H. JR., BORNING, A.

2008 *Value Sensitive Design and Information Systems*, in HIMMA, K. E., TAVANI, H. T. (eds.), *The Handbook of Information and Computer Ethics*, John Wiley & Sons, Hoboken (NJ), pp. 69–101.

FRIXIONE, M., PALLADINO, D.

2011 *La computabilità: algoritmi, logica, calcolatori*, Carocci, Roma.

FUTURE OF LIFE INSTITUTE

2023 *Pause Giant AI Experiments: An Open Letter*, 22 marzo 2023.

GABBRIELLI, M.; MARTINI, S.

2011 *Linguaggi di programmazione. Principi e paradigmi*. McGraw-Hill, Milano.

GALISON, P.

1993 *The Cultural Meaning of Aufbau*, in F. Stadler (ed.), *Scientific Philosophy: Origins and Developments*, Springer, Dordrecht, pp. 75-93.

1994 *The Ontology of the Enemy: Norbert Wiener and the Cybernetic Vision*, *Critical Inquiry* 21 (1):228-266

(1994) 10.1086/448747, in (a cura di) Bianchini, F., Franchi, S., Matteuzzi, M., *Verso un'archeologia dell'intelligenza artificiale*, In "Discipline Filosofiche", XVII I, Quodlibet, Macerata, 2007, pp.41-84.

1996 *Computer Simulations and the Trading Zone*, in P. L. Galison e D. J. Stump (a cura di), *The Disunity of Science: Boundaries, Contexts, and Power*. Stanford University Press, Stanford (CA), pp. 118–157.

GARDNER, H.

1985 *The Mind's New Science: A History of the Cognitive Revolution*, Basic Books; trad. it. *La nuova scienza della mente: Storia della rivoluzione cognitiva*, Feltrinelli, Milano, 2016.

GATES, B.

1976 *An Open Letter to Hobbyists*. *Homebrew Computer Club Newsletter*, Vol. 2, No. 1, p. 2.

GEROVITCH, S.

2002 *From Newspeak to Cyberspeak: A History of Soviet Cybernetics*, MIT Press, Cambridge (MA). ISBN (electronic): 9780262273688. <https://doi.org/10.7551/mitpress/3137.001.0001>

GIBSON, W.

1984 *Neuromancer*, Ace Books, New York; trad. it. *Neuromante*, Nord, Milano, 1986.

GIGERENZER, G.

2022 *How to Stay Smart in a Smart World: Why Human Intelligence Still Beats Algorithms*, MIT Press; trad. it. *Perché l'intelligenza batte gli algoritmi*, Cortina, Milano, 2023.

GLEICK, J.

2011 *The Information: A History, a Theory, a Flood*, Pantheon Books, New York; trad. it. *Informazione. Una storia, una teoria, un diluvio*, Feltrinelli, Milano, 2012.

GÖDEL, K.

1931 *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I*, Monatshefte für Mathematik und Physik, 38, pp. 173–198.

GOLDBERG, D.

1991 – *What Every Computer Scientist Should Know About Floating-Point Arithmetic*, *ACM Computing Surveys*, 23(1), pp. 5-48.

GOLDSTINE, H.H.

1972 *The Computer from Pascal to von Neumann*, 1972, Princeton University Press, Princeton.

GOLDSTINE A, GOLDSTINE H.

1946 "The Electronic Numerical Integrator and Computer (ENIAC)," in *IEEE Annals of the History of Computing*, vol. 18, no. 1, pp. 10-16, Spring 1996, doi: 10.1109/85.476557.

GOLDSTINE, H. H.; VON NEUMANN, J.

1947 *Planning and Coding Problems for an Electronic Computing Instrument*, Part II, Vol. 1, Report prepared for U.S. Army Ordnance Dept., in *John von Neumann Collected Works*, Vol. V, a cura di A. H. Taub, Pergamon Press, 1963, pp. 80–151.

GOOD, J.; MICHIE, D.; TIMMS, G.

1945 *General General Report on Tunny with Emphasis on Statistical Methods*, in ed. Reeds, J.A, Diffie, W., Field, J.V., *Breaking Teleprinter Ciphers at Bletchley Park*, John Wiley & Sons, Hoboken (NJ) 2015.

GOOD, I.J.

1980 *Pioneering Work on Computers at Bletchley*, in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.31-46.

2006 *From Hut 8 to Newmanry* in ed. Copeland, B. J, and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.204-222.

GOODFELLOW, I., BENGIO, Y., COURVILLE, A.

2016 *Deep Learning*, MIT Press, Cambridge (MA). <http://www.deeplearningbook.org>

GOODMAN, N.

1962 – *Languages of Art: An Approach to a Theory of Symbols*, Bobbs-Merrill Company, New York.

GRAY, C. M., KOU, Y., BATTLES, B., HOGGATT, J., TOOMBS, A. L.

2018 *The Dark (Patterns) Side of UX Design*, in *CHI '18: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, Paper No. 534, pp. 1–14. <https://doi.org/10.1145/3173574.3174108>

GROSSBERG, S.

1976 *Adaptive Pattern Classification and Universal Recoding: II. Feedback, Expectation, Olfaction, Illusions*, *Biological Cybernetics*, 23, pp. 187–202. <https://doi.org/10.1007/BF00340335>

GRUDIN, J.

2017 *From Tool to Partner: The Evolution of Human-Computer Interaction*, Morgan & Claypool (Synthesis Lectures on Human-Centered Informatics).

<https://www.morganclaypool.com/doi/abs/10.2200/S00745ED1V01Y201612HCI035>

GUGERTY, L.

2006 *Newell and Simon's Logic Theorist: Historical Background and Impact on Cognitive Modeling*, *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9), 880-884.

<https://doi.org/10.1177/154193120605000904>

HALL, E. C.

1996 *Journey to the Moon: The History of the Apollo Guidance Computer*. *Library of Flight Series*. Reston (VA), AIAA.

HAIGH, T.

2001 *Inventing Information Systems: The Systems Men and the Computer, 1950–1968*, *Business History Review*, 75(1), pp. 15–61. <https://www.jstor.org/stable/3116556>.

HAIGH, T, CERUZZI, P.E.

2021 *A New History of Modern Computing*, MIT Press, Cambridge, London.

HAIGH, T.; PRIESTLEY, M.; ROPE, C.

2014a *Engineering "The Miracle of the ENIAC": Implementing the Modern Code Paradigm*. *IEEE Annals of the History of Computing*, 36(2), pp. 41–59. doi: 10.1109/MAHC.2014.15.

2014b *Los Alamos Bets on ENIAC: Nuclear Monte Carlo Simulations, 1947–1948*. *IEEE Annals of the History of Computing*, 36(3), pp. 42–63. doi: 10.1109/MAHC.2014.40.

2016 *ENIAC in Action Making and Remaking the Modern Computer*, MIT Press, Cambridge London.

HARAWAY, D.

1985 *A Manifesto for Cyborgs: Science, Technology, and Socialist Feminism in the 1980s*, Socialist Review; trad. it. *Manifesto cyborg: Donne, tecnologie e biopolitiche del corpo*, Feltrinelli, Milano 2018.

HARIRI, R. H.; FREDERICKS, E. M.; BOWERS, K. M.

2019 *Uncertainty in Big Data Analytics: Survey, Opportunities, and Challenges*. *Journal of Big Data*, 6, 44 (2019). <https://doi.org/10.1186/s40537-019-0206-3>

HARNAD, S.

1990 *The Symbol Grounding Problem*, in *Physica D* 42, pp. 335-346.

HARTLEY, R.

1928 *Trasmission of informations*, in *Bell Sistem Technical Journal* (July) 7, pp. 535-563.

HARTREE, D.R.

1940 *"The Bush Differential Analyser and Its Applications."* *Nature*, 146, pp. 319-323

HAUGELAND, J.

1985 *Artificial Intelligence: The Very Idea*, MIT Press, Cambridge Massachusetts; trad. it. *L'Intelligenza artificiale*, Bollati Boringhieri, Torino, 1988.

HAYLES, N. K.

1999 *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics*, The University of Chicago Press, Chicago–London.

HEBB, D. O.

1949 *The Organization of Behavior. A Neuropsychological Theory*, Wiley, New York.

HEAVISIDE, O.

1893 – *Electromagnetic Theory* (Vol. 1), Benn Brothers, London.

HEIMS, S.

1980 *John Von Neumann and Norbert Weiner. From Mathematics to the Technologies of Life and Death*, MIT press, Cambridge MA.

1991 *The Cybernetics Group*, MIT Press; trad. it. *I cibernetici: Storia di un gruppo scientifico*, Editori Riuniti, Roma, 1993.

HILBERT, D.

1902 *Mathematical problems*, *Bull. Amer. Math. Soc.* 8(10): 437-479 (July 1902).

HILBERT, D., ACKERMANN, W.

1928 *Grundzüge der theoretischen Logik*, Berlin: Springer.

HILL, M.

2004 – *Bletchley Park People: Churchill's Geese that Never Cackled*, Sutton Publishing Ltd, Gloucester.

HILL, R. K.

2015 *What an Algorithm Is*, *Philosophy & Technology*, 28(3), pp. 313–328. DOI: 10.1007/s13347-014-0184-5.

- HILLIS, W. D.
1984 *The Connection Machine: A Computer Architecture Based on Cellular Automata*, *Physica D: Nonlinear Phenomena*, 10(1-2), pp. 213-228. [https://doi.org/10.1016/0167-2789\(84\)90263-X](https://doi.org/10.1016/0167-2789(84)90263-X)
- 1986 *The Connection Machine*, MIT Press, Cambridge (MA).
- HINTON, G. E., SEJNOWSKI, T. J.
1983 *Optimal Perceptual Inference*, in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, New York, pp. 448-453.
- HINTON, G. E., OSINDERO, S., TEH, Y. W.
2006 *A Fast Learning Algorithm for Deep Belief Nets*, *Neural Computation*, 18, pp. 1527-1554.
- HOFSTADTER, D. R.
1979 *Gödel, Escher, Bach: an eternal golden braid*, Basic Books; trad. It. *Gödel, Escher, Bach: un'eterna ghirlanda brillante*, Adelphi, Milano 1990
- HODGES, A.
1983 *Alan Turing, The Enigma*, Princeton University Press, Princeton NJ; trad. it. *Alan Turing, storia di un enigma*, Bollati Boringhieri, Torino, 2014.
- HOGARTH, I.
2023 *We Must Slow Down the Race to God-like AI*, *Financial Times*, 13 aprile 2023.
- HOEFLER, D. C.
1971 *Silicon Valley, USA*. In *Electronic News*. Rist. in *The Evolution of Silicon Valley*. Mountain View (CA), Computer History Museum.
- HOEPMAN, J.-H.
2021 *Privacy Is Hard and Seven Other Myths: Achieving Privacy through Careful Design*, MIT Press, Cambridge (MA).
- 2022 *Privacy Design Strategies (The Little Blue Book)*. <https://creativecommons.org/licenses/by-nc/4.0/>
- HOLBROOK, D.
2009 *Controlling Contamination: The Origins of Clean Room Technology*. *History and Technology*, 25(3), pp. 173-191. <https://doi.org/10.1080/07341510903083203>.
- HOLLAND, J. H.
1962 *Outline for a Logical Theory of Adaptive Systems*. *Journal of the ACM*, 9(3), pp. 297-314.
- 1975 *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor; 2^a ed. MIT Press, 1992.
- HOLLERITH, H.
1890 *An Electric Tabulating System*, in Randell, B. *The Origins of Digital Computers Selected Papers*, 2 ed. Springer-Verlag Berlin Heidelberg New York 1975, pp.129-139.
- HOPPER MURRAY, G.
1952 *The Education of a Computer*. Remington Rand Corp. Presentato all'Association for Computing Machinery, 2-3 maggio 1952.
- 1981 *Keynote Address*, in Wexelblat, R. L. (a cura di), *History of Programming Languages*. Academic Press, Los Angeles, pp. 7-24.
- HOROWITZ, P.; HILL, W.
2015 *The Art of Electronics* (3rd ed.). Cambridge, Cambridge University Press.
- HORWOOD, D. C.

- 1973 *A Technical Description of COLOSSUS I*. Rapporto tecnico del Post Office Research Station, Dollis Hill, agosto 1973 (declassificato nel 2003).
- HUANG, L., YU, W., MA, W., ZHONG, W., FENG, Z., WANG, H., ... LIU, T.
2025 *A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions*, *ACM Transactions on Information Systems*, 43(2), pp. 1–55.
- HUSKEY, H.D.
2003 *Williams Tube Memory*, in Ralston, A.; Reilly, E. D. (eds.), *Encyclopedia of Computer Science*, 4^a ed., John Wiley & Sons Ltd., Chichester, gennaio 2003, pp. 1851–1853.
- 2005 *The state of the art in electronic digital computing in Britain and the United States (1947)*, in in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.529-540.
- HUSKEY, H. D., HUSKEY, V. R.
1980 *Lady Lovelace and Charles Babbage*, *Annals of the History of Computing*, Vol. 2, No. 4, pp. 299-329.
- HYMAN, A.
1982 *Charles Babbage: Pioneer of the Computer*, Princeton University Press.
- IEEE COMPUTER SOCIETY
2008–2019 – *IEEE Standard for Floating-Point Arithmetic*, IEEE Std 754-2008; revised as IEEE Std 754-2019, IEEE, New York.
- IEMHOFF, R.
2019 *Intuitionism in the Philosophy of Mathematics*, The Stanford Encyclopedia of Philosophy (Fall 2019 Edition).
- IFRAH, G.
1998 – *The Universal History of Numbers: From Prehistory to the Invention of the Computer*, Harvill Press, London.
- ILDIZ, M. E., HUANG, Y., LI, Y., RAWAT, A. S., OYMAK, S.
2024 *From Self-Attention to Markov Models: Unveiling the Dynamics of Generative Transformers*, *arXiv preprint*.
<https://doi.org/10.48550/arXiv.2402.13512>
- IRE Transactions on Information Theory
1956 *1956 Symposium on Information Theory*, *IRE Transactions on Information Theory*, Vol. IT-2, No. 3, settembre 1956. Atti del *1956 Symposium on Information Theory* tenutosi al *Massachusetts Institute of Technology*, Cambridge (MA), 10–12 settembre 1956.
- IRELAND, E
2017 *We were the world's first computer operators*, in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.161-166.
- ISRAEL, G.; MILLÁN GASCA, A.
2008 *Il mondo come gioco matematico. La vita e le idee di John von Neumann*. Bompiani, Milano.
- IVAKHNENKO, A. G., LAPA, V. G.
1966 *Cybernetic Predicting Devices*, CCM Information Corp., New York (trad. ingl. dell'originale sovietico sul Group Method of Data Handling – GMDH).
- JACOBS, J. F.
1983 *SAGE Overview*. *IEEE Annals of the History of Computing*, 5(4), pp. 323–329.
<https://doi.org/10.1109/MAHC.1983.10101>.

- JEFFRESS, L. A. (a cura di)
1951 *Cerebral Mechanisms in Behavior: The Hixon Symposium*, John Wiley & Sons, New York.
- JENNINGS, H.S.
1906 *Behavior of the lower organisms*. Columbia University Press. <https://doi.org/10.1037/10817-000>
1910 *Diverse ideals and divergent conclusions in the study of behavior in lower organisms*, in *American Journal of Psychology*, 21, pp. 349-370.
- JOHNSON, D. G.
1985 *Computer Ethics*. Prentice-Hall, Englewood Cliffs (NJ).
1994 *Computer Ethics* (2^a ed.). Prentice-Hall, Englewood Cliffs (NJ).
- JOHNSON, D. G.; MILLER, K.
2009 *Computer Ethics: Analyzing Information Technology* (4th ed.). Upper Saddle River (NJ), Pearson Education International.
- JOHNSON, D. G., WETMORE, J. M.
2021 *Technology and Society: Building Our Sociotechnical Future* (2^a ed.), MIT Press, Cambridge (MA).
- KAHN, D.
1967 *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet*, Macmillan.
- KAHN, H.
1955 *Use of Different Monte Carlo Sampling Techniques*. RAND Corporation, Report P-766, Santa Monica, 30 novembre.
- KANAL, L.
1962 *Evaluation of a Class of Pattern-Recognition Networks*, in BERNARD, E. E., KARE, M. R. (a cura di), *Biological Prototypes and Synthetic Systems*, Plenum Press, New York, pp. 261-269
- KARI, J.
2005 *Theory of Cellular Automata: A Survey*. *Theoretical Computer Science*, 334, pp. 3-33.
- KEMENY, J. G.
1955 *Man Viewed as a Machine*. *Scientific American*, April.
- KENNEDY, T. R. Jr.
1946 *Electronic Computer Flashes Answers, May Speed Engineering*, in *New York Times*, 15 febbraio 1946.
- KEYNES, J. M.
1921 *A Treatise on Probability*. Macmillan and Co., London.
- KILBY, J. S.
1964 *Miniaturized Electronic Circuits*. US Patent 3138743, published 23 June 1964. Filed 6 February 1959.
- KIM, K.; KOH, G.
2004 *The Prospect on Semiconductor Memory in the Nano Era*, in *Proceedings of the IEEE 7th International Conference on Solid-State and Integrated-Circuit Technology (ICSICT)*, vol. 1, 2004, pp. 662-667.
- KLEENE, S. C.
1954 – *Representation of Events in Nerve Nets and Finite Automata*, in C. SHANNON & J. MCCARTHY (a cura di), *Automata Studies*, Princeton University Press, Princeton.
- KNUTH, D. E.; TRABB PARDO, L.
1980 *The Early Development of Programming Languages*, in Metropolis, N.; Howlett, J.; Rota, G.-C. (eds.), *A History of Computing in the Twentieth Century*, Academic Press, New York, pp. 197-273.

- KESEY, K.
1962 *One Flew Over the Cuckoo's Nest*, Signet, New York; trad. it.; *Qualcuno volò sul nido del cuculo*, Rizzoli, 1988.
- KLINE, R.
2015 *The Cybernetics Moment: Or Why We Call Our Age the Information Age*, Johns Hopkins University Press.
- KNEALE, W., KNEALE, M.
1962 *The Development of Logic*, Clarendon Press; trad. it. *Storia della logica*, Einaudi, Torino.
- KOHONEN, T.
1972 *Correlation Matrix Memories*, *IEEE Transactions on Computers*, C-21(4), pp. 353–359.
- KRIZHEVSKY, A., SUTSKEVER, I., HINTON, G. E.
2012 *ImageNet Classification with Deep Convolutional Neural Networks*, *Advances in Neural Information Processing Systems*, 25(2). <https://doi.org/10.1145/3065386>
- KURZWEIL, R.
1990 *The Age of Intelligent Machines*, MIT Press.

1999 *The Age of Spiritual Machines: When Computers Exceed Human Intelligence*, Viking.

2005 *The Singularity Is Near: When Humans Transcend Biology*, Viking.
- LANEY, D.
2001 *3D Data Management: Controlling Data Volume, Velocity and Variety*. META Group Research Note, 6.
- LANGTON, C. G. (ed.)
1988 *Artificial Life: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems*. Addison-Wesley.
- LANIER, J.
2010 *You Are Not a Gadget: A Manifesto*, A. Knopf, New York; trad. it. *Tu non sei un gadget*, Mondadori, Milano, 2010.

2017 *Dawn of the New Everything: Encounters with Reality and Virtual Reality*, Henry Holt and Co., New York; trad. it. *L'alba del nuovo tutto. Il futuro della realtà virtuale*, IL Saggiatore, Milano 2019.
- LASHLEY, K. S.
1951 *The Problem of Serial Order in Behavior*, in JEFFRESS, L. A. (a cura di), *Cerebral Mechanisms in Behavior: The Hixon Symposium*, John Wiley & Sons, New York, pp. 112–146.
- LARSSON, E.J.
2021 *The Myth of Artificial Intelligence: Why Computers Can't Think the Way We Do*; trad. it. *Il mito dell'intelligenza artificiale. Perché i computer non possono pensare come noi*, Franco Angeli, Milano, 2022.
- LATOUR, B.
1991 *Nous n'avons jamais été modernes. Essai d'anthropologie symétrique*, La Découverte, Parigi; trad. it. *Non siamo mai stati moderni. Saggio d'antropologia simmetrica*, Elèuthera, Milano, 2009.
- LAVINGTON S.H.
1980 *Computer Development at Manchester University* in in Metropolis, N.; Howlett, J.; Rota, G.-C. (eds.), *A History of Computing in the Twentieth Century*, Academic Press, New York, pp. 433–444.
- LE DANTEC, C. A., POOLE, E. S., WYCHE, S. P.
2009 *Values as Lived Experience: Evolving Value Sensitive Design in Support of Value Discovery*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, New York (NY), pp. 1141–1150.
<https://doi.org/10.1145/1518701.1518875>

- LECUN, Y.
1985 *A Learning Procedure for Asymmetric Threshold Networks*, *Proceedings of Cognitiva 85*, Paris, pp. 599–604.
- 1988 *A theoretical framework for back-propagation*. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School*, pages 21–28, CMU, Pittsburgh, Pa, 1988. Morgan Kaufmann.
- LECUN, Y., BENGIO, Y., HINTON, G.
2015 *Deep Learning*, *Nature*, 521, pp. 436–444. <https://doi.org/10.1038/nature14539>
- LECUYER, C.; BROCK, D. C.; LAST, J.
2010 *Makers of the Microchip: A Documentary History of Fairchild Semiconductor*. Cambridge (MA), MIT Press.
- LEE K., CHEN, Q.
2021 *AI 2041. Ten Visions for Our Future*, Penguin, New York; trad. it. *AI 2041. Scenari dal futuro dell'intelligenza artificiale*, Luiss University Press, Roma, 2023.
- LEHTO, O.
1998 *Mathematics without borders: a history of the International Mathematical Union*, Springer-Verlag.
- LEIBNIZ, G. W.
1703 – *Explication de l'arithmétique binaire, où il se sert des seuls caractères 0 et 1, avec des remarques sur son utilité, et sur qu'elle donne le sens des anciennes figures chinoises de Fohy*, Académie royale des sciences, Paris.
- LEM, S.
1968 *Głos Pana*. Warszawa: Wydawnictwo Literackie; trad. ingl. *His Master's Voice*, Harcourt, Brace & World, New York 1983; trad. it. *La voce del padrone*, Mondadori, Milano 2022.
- LEONARD, R.
2010 *Von Neumann, Morgenstern, and the Creation of Game Theory*, Cambridge University Press.
- LEPORE, J.
2018 'What 2018 Looked Like Fifty Years Ago', *The New Yorker*, 31 December, trad. it. in Nowotny, H., 2019 *Le macchine di Dio: Come la tecnologia plasma il nostro futuro*, LUISS, Roma.
- LEVY, S.
1984 *Hackers: Heroes of the Computer Revolution*, Anchor Press/Doubleday
- 1992 *Artificial Life: The Quest for a New Creation*. Pantheon Books, New York.
- LI, S.; XU, L. D.; ZHAO, S.
2015 – *The Internet of Things: A Survey*, *Information Systems Frontiers*, 17(2), pp. 243-259. <https://doi.org/10.1007/s10796-014-9492-7>
- LIAO, S. M.
2020 *A Short Introduction to the Ethics of Artificial Intelligence*, in S. Matthew Liao (ed.), *Ethics of Artificial Intelligence*, Oxford University Press, pp.1-42.
- LICKLIDER, J. C. R.
1950 *The Manner in Which and Extent to Which Speech Can Be Distorted and Remain Intelligible*. In: PIAS, C. (ed.), *Cybernetics. The Macy Conferences 1946–1953, The Complete Transactions*. Zürich–Berlin, diaphanes, 2016, pp. 203–247.
- 1960 *Man-Computer Symbiosis*. *IRE Transactions on Human Factors in Electronics*, HFE-1(1), pp. 4–11. doi:10.1109/THFE2.1960.4503259.
- 1963 *Topics for Discussion at the Forthcoming Meeting. Memorandum for: Members and Affiliates of the Intergalactic Computer Network*. Washington (DC), Advanced Research Projects Agency, 23 April.

- 1965 *Libraries of the Future*. Cambridge (MA), MIT Press.
- LICKLIDER, J. C. R.; TAYLOR, R.W.
1968 *The Computer as a Communication Device*. *Science and Technology*, April.
- LOLLI, G.
1992 *Incompletezza. Saggio su Kurt Gödel*, Il Mulino, Bologna.
- 2011 *La guerra dei trent'anni (1900-1930). Da Hilbert a Gödel*, ETS, Pisa.
- 2016 *Tavoli, sedie, boccali di birra: David Hilbert e la matematica del Novecento*, Cortina, Milano. 2019 *I teoremi di incompletezza*, Il Mulino, Bologna.
- LUBAR, S.
1992 *Do Not Fold, Spindle or Mutilate: A Cultural History of the Punch Card*, in «Journal of American Culture», vol. 15, n. 4, pp. 43-55.
- LUCAS, J. R.
1961 *Minds, Machines and Gödel*, in *Philosophy*, Apr. - Jul., 1961, Vol. 36, No. 137 (Apr. - Jul., 1961), pp. 112-127.
- LUNDGREN, B.
2017 *Does Semantic Information Need to Be Truthful? Synthese*, 196(7), Special Issue on *Between First- and Second-Order Logic* (July 2019), pp. 2885–2906.
- MACASKILL, W.
2017 *Effective Altruism: Introduction, Essays*, Oxford.
- 2022 *What We Owe the Future*, Oneworld, London; trad. it. *Ciò che dobbiamo al futuro*, Einaudi, Milano, 2023.
- MAGNELLO, E.
2005 *The National Physical Laboratory*, in COPELAND, B. J. et al., *Alan Turing's Electronic Brain: The Struggle to Build the ACE, the World's Fastest Computer*, MIT Press, Cambridge (MA), pp. 47–92.
- MAHON, P.
1945 *History of Hut 8 to December 1941*, in COPELAND, B. J. (a cura di), *The Essential Turing*, Oxford University Press, Oxford, pp. 265–312.
- MANGIONE, R., RIZZI, A.
1993 *Storia della logica: Da Boole ai nostri giorni*, FrancoAngeli, Milano.
- MARCONI, D.
2025 *L'intelligenza di Turing*, in MARCONI, A. (a cura di), *M. Turing. Macchine Calcolatrici e Intelligenza*, Einaudi, Milano.
- MARCUS, G.
2018 *Deep Learning: A Critical Appraisal*, *arXiv*, Vol. 1801.00631. <http://dx.doi.org/arXiv:1801.00631>.
- MARKOFF, J.
2003 *The New York Times*, 22 febbraio 2003.
- 2005 *What the Dormouse Said: How the Sixties Counterculture Shaped the Personal Computer*, Penguin, New York.
- MARR, D.
1982 *Vision. A Computational Investigation into the Human Representation and Processing of Visual Information*, MIT Press, Cambridge (MA); ed. 2010.

- MARTIN, C. D.
1995 *ENIAC: The Press Conference That Shook the World*, *IEEE Technology and Society Magazine*, dicembre 1995.
- MASANI, P. R.
1987 *Norbert Wiener and the Computer*, *Annals of the History of Computing*, 9, pp. 183–197.
1990 *Norbert Wiener, 1894–1964*, Birkhäuser, Basel.
1990b *McCulloch, Pitts and the Evolution of Wiener's Neurophysiological Ideas*, in *Norbert Wiener 1894–1964*, pp. 218–238.
- MASANI, P. (a cura di)
1979–1985 *Norbert Wiener: Collected Works with Commentaries*, 4 voll., MIT Press, Cambridge (MA).
- MASANI, P., PHILLIPS, R. S.
1985 *Antiaircraft Fire-Control and the Emergence of Cybernetics*, in MASANI, P. (a cura di), *Norbert Wiener: Collected Works with Commentaries*, 4 voll., MIT Press, Cambridge (MA), 1979–1985.
- MAUCHLY, J. W.
1942 *The Use of High Speed Vacuum Tube Devices for Calculating*, in RANDELL, B. (a cura di), *The Origins of Digital Computers: Selected Papers* (2^a ed.), Springer-Verlag, Berlin–Heidelberg–New York, 1975, pp. 329–332.
1980 *The ENIAC*, in METROPOLIS, N. et al. (a cura di), *A History of Computing in the Twentieth Century*, Academic Press, New York–London, pp. 541–550.
1999 *ENIAC: The Triumphs and Tragedies of the World's First Computer*, Walker and Co., New York.
- MAUCHLY J. – ECKERT J. P.
1945 *Automatic High Speed Computing: A Progress Report on the EDVAC*, Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia, 30 settembre 1945.
- MAXWELL, J. CLERK
1872 *Theory of Heat*, 3^a ed., Longmans, Green and Company, London (ristampa Greenwood Press, 1970).
- MCCARTHY, J.
1960 *Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I*, *Communications of the ACM*, 3(4), pp. 184–195. <https://doi.org/10.1145/367177.367199>.
1981 *History of LISP*, in WEXELBLAT, R. L. (a cura di), *History of Programming Languages*, Academic Press, Los Angeles, pp. 173–197.
- MCCARTHY, J.; HAYES, P.
1969 *Some Philosophical Problems from the Standpoint of Artificial Intelligence*, in MELTZER, B., MICHIE, D. (a cura di), *Machine Intelligence 4*, Edinburgh University Press, Edinburgh, pp. 463–502.
- MCCARTHY, J., MINSKY, M. L., ROCHESTER, N., SHANNON, C. E.
1956 *A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence*, Dartmouth College, Hanover (NH).
- MCCARTNEY, S.
1999 *ENIAC: The Triumphs and Tragedies of the World's First Computer*, Walker, New York.
- MCCLELLAND, J. L., RUMELHART, D. E.
1981 *An Interactive Activation Model of Context Effects in Letter Perception: I. An Account of Basic Findings*, *Psychological Review*, 88(5), pp. 375–407. <https://doi.org/10.1037/0033-295X.88.5.375>.
1986 *On Learning the Past Tenses of English Verbs*, in RUMELHART, D. E., MCCLELLAND, J. L. (a cura di), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 2, MIT Press, Cambridge (MA), pp. 216–271.

MCCLELLAND, J. L., RUMELHART, D. E., HINTON, G.

1986 *The Appeal of Parallel Distributed Processing*, in RUMELHART, D. E., MCCLELLAND, J. L. (a cura di), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, MIT Press, Cambridge (MA), pp. 3–44; trad. it. *Il fascino del PDP*, in RUMELHART, D. E., MCCLELLAND, J. L., *PDP. Microstruttura dei processi cognitivi*, Il Mulino, Bologna, 1991, pp. 31–80.

MCCORDUCK, P.

2004 *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence*, 2^a ed., A. K. Peters/CRC Press, New York.

MCCULLOCH, W. S.

1974 *Recollections of the Many Sources of Cybernetics*, in BUCK, R. C., COHEN, R. S. (a cura di), *PSA 1972: Proceedings of the Biennial Meeting of the Philosophy of Science Association*, Springer, Dordrecht, pp. 19–32.

MCCULLOCH, W. S., PITTS, W.

1943 *A Logical Calculus of the Ideas Immanent in Nervous Activity*, *Bulletin of Mathematical Biophysics*, 5, pp. 115–133; trad. it. in *Filosofia degli automi: Dalla cibernetica all'intelligenza artificiale*, Bollati Boringhieri, Torino.

MCCULLOCH, W. S., PFEIFFER, J.

1949 *Of Digital Computers Called Brains*, *Scientific Monthly*, 69, pp. 368–376.

MEDLER, D. A.

1998 *A Brief History of Connectionism*, *Neural Computing Surveys*, 1, pp. 61–101.

MEEK, C. L.

2004 *Mainframe*, in REILLY, E. D. (a cura di), *Concise Encyclopedia of Computer Science*, John Wiley & Sons, Hoboken (NJ).

MENABREA, L.

1842 *Sketch of the Analytical Engine Invented by Charles Babbage*, *Bibliothèque Universelle de Genève*, No. 82.

METROPOLIS, N.

1980 *The MANIAC: Computer Development at the Institute for Advanced Study*, in METROPOLIS, N. et al. (a cura di), *A History of Computing in the Twentieth Century*, Academic Press, New York–London, pp. 457–464.

1987 *The Beginning of the Monte Carlo Method*, *Los Alamos Science*, 15, pp. 125–130.

1992 *The Age of Computing: A Personal Memoir*, *Daedalus*, 121(1), pp. 119–130.

METROPOLIS, N.; REITWIESNER, G.; VON NEUMANN, J.

1950 *Statistical Treatment of Values of First 2000 Decimal Digits of e and of π Calculated on the ENIAC*, *Mathematical Tables and Other Aids to Computation*, 4, pp. 109–111; anche in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 765–767.

METROPOLIS, N.; ULAM, S.

1949 *The Monte Carlo Method*, *Journal of the American Statistical Association*, 44, pp. 335–341.

MICHIE, D.

2006 *Codebreaking and Colossus*, in COPELAND, B. J. et al. (a cura di), *Colossus: The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford, pp. 204–222.

MILLER, G. A.

1956 *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information*, *Psychological Review*, 63(2), pp. 81–97.

MINSKY, M., PAPERT, S.

1969 *Perceptrons: An Introduction to Computational Geometry*, MIT Press, Cambridge (MA).

- MITCHELL, M.
2019 *Artificial Intelligence: A Guide for Thinking Humans*, Farrar, Straus and Giroux, New York; trad. it. *L'intelligenza artificiale. Una guida per esseri umani pensanti*, Einaudi, Torino, 2022.
- MITCHELL, T.
1997 *Machine Learning*, McGraw-Hill, New York.
- MOHRI, M., ROSTAMIZADEH, A., TALWALKAR, A.
2022 *Foundations of Machine Learning*, 2^a ed., MIT Press, Cambridge (MA).
- MOOR, J. H.
1976 *An Analysis of the Turing Test*, *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition*, 30(4), pp. 249–257.
2001 *The Status and Future of the Turing Test*, *Minds and Machines*, 11, pp. 77–93.
2003 (a cura di) *The Turing Test: The Elusive Standard of Artificial Intelligence*, Kluwer, Dordrecht.
- MOORE, G.
MOORE, G. E.
1965 *Cramming More Components onto Integrated Circuits*, *Electronics*, 38(8), 19 aprile 1965, pp. 114 ss.
1976 *Microprocessors and Integrated Electronic Technology*, *Proceedings of the IEEE*, 64(6), pp. 837–841.
- MORI, M.
1970 *Bukimi no tani – The Uncanny Valley* (K. F. MacDorman & T. Minato, trad.), *Energy*, 7(4), pp. 33–35.
- MORABITO, C.
2006 *La mente nel cervello. Un'introduzione storica alla neuropsicologia cognitiva* (3^a ed.), Laterza, Bari.
- MYERS, B. A.
1996 *A Brief History of Human-Computer Interaction Technology*, *ACM Interactions*, 5(2), marzo 1998, pp. 44–54.
- MYHILL, J.
1963 *The Converse of Moore's Garden-of-Eden Theorem*, *Proceedings of the American Mathematical Society*, 14(4), pp. 685–686. <https://doi.org/10.1090/S0002-9939-1963-0155764-9>.
- NAGEL, E., NEWMAN, J. R.
1958 *Gödel's proof*, New York: New York University Press; trad. It. *La prova di Gödel*, Bollati Boringhieri Torino, 2 ed. 1992.
- NAHIN, P. J.
2013 *The Logician and the Engineer: How George Boole and Claude Shannon Created the Information Age*, Princeton University Press; trad. it. *Il logico e l'ingegnere: L'alba dell'era digitale*, Codice Torino, 2017.
- NAPPER, B.
2010 *Max Newman and the Mark 1. Computer50*, 11 May 2008. Web. 30 Jan. 2010. <https://web.archive.org/web/20080511182631/http://www.computer50.org/mark1/newman.html>
- NEWBORN, M.
1997 *Kasparov vs. Deep Blue: Computer Chess Comes of Age*, Springer-Verlag, Berlin–Heidelberg.
- NEWMAN, M.
1943 Minuta a E. W. H. Travis 1 marzo 1943 e Minuta a E. W. H. Travis 13 marzo 1943 in Reeds, J.A *Appendix D: Initial Conception of Colossus* in in ed. Reeds, J.A, Diffie, W., Field, J.V., *Breaking Teleprinter Ciphers at Bletchley Park*, John Wiley & Sons, Hoboken (NJ) 2015, pp.535-539,
1948 *A Status Report on the Royal Society Computing Machine Laboratory*, prepared by Professor Newman for an internal committee meeting of the University of Manchester, Oct. 15th 1948, in Napper, B., 2010, *Max Newman*

and the Mark 1. *Computer50*, 11 May 2008. Web. 30 Jan. 2010.

<https://web.archive.org/web/20080511182631/http://www.computer50.org/mark1/newman.html>

NEWMAN, W.

2006 *Max Newman—Mathematician, Codebreaker, and Computer Pioneer* in ed. Copeland, B. J. and Others, *Colossus, The Secrets of Bletchley Par's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.176-188.

NEWELL, A., SIMON, H. A.

1956 *The Logic Theory Machine: A Complex Information Processing System*, RAND Corporation, P-868.

1956b *The Logic Theory Machine* in *1956 Symposium on Information Theory, IRE Transactions on Information Theory*

1976 *Computer Science as Empirical Inquiry: Symbols and Search*, Communications of the ACM, 19(3), 113–126. <https://doi.org/10.1145/360018.360022> In J. Haugeland (a cura di), *Mind Design II: Philosophy, Psychology, and Artificial Intelligence* Cambridge, MA: MIT Press. 1997, pp. 81–109).

NICOD, J. G. P.

1917 *A Reduction in the Number of the Primitive Propositions of Logic. Proceedings of the Cambridge Philosophical Society*, 19, pp. 32–41.

NORMAN, D. A.

1988 *The Psychopathology of Everyday Things [The Design of Everyday Things]*, Basic Books, New York; trad. it. *La caffettiera del masochista. Psicopatologia degli oggetti quotidiani*, Giunti, Firenze, 1996.

1991 *Cognitive Artifacts*, in CARROLL, J. M. (ed.), *Designing Interaction: Psychology at the Human-Computer Interface*, Cambridge University Press, Cambridge, pp. 17–38.

1998 *The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution*, MIT Press, Cambridge (MA).

NOWOTNY, H.

2019 *Le macchine di Dio: Come la tecnologia plasma il nostro futuro*, LUISS, Roma.

NOYCE, R. N.

1961 *Semiconductor Device-and-Lead Structure. US Patent 2981877*, filed 30 July 1959, granted 25 April 1961.

NUMERICO, T.

2021 *Big Data e algoritmi, Prospettive critiche*. Carocci, Roma

2005 *From Turing Machine to “electronic brain”*, in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.173-192.

NUMERICO, T.; FIORMONTE, D.; TOMASI, F.

2010 *L'umanista digitale*. Bologna, Il Mulino.

NYQUIST, H.

1924 *Certain Factors Affecting Telegraph Speed*, Bell System Technical Journal, Vol.3, Issue 2, pp.324-346.

O'KEEFE, S. E. M., & EDWARDS, J.

2016 *Eager recirculating memory to alleviate the von Neumann Bottleneck*. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 1-5). IEEE.

O'NEIL, C.

2016 *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*, Crown Publishing, New York; trad. it. *Armi di distruzione matematica*, Bompiani, Milano 2017.

O'NEIL, C., GUNN, H.

2020 *Near-Term Artificial Intelligence and the Ethical Matrix*, in S. Matthew Liao (ed.), *Ethics of Artificial Intelligence*, Oxford University Press, pp. 237-270.

OLAZARAN, M.

1996 *A Sociological Study of the Official History of the Perceptrons Controversy*, *Social Studies of Science*, 26(3), pp. 611-659.

OPPY, G., DOWE, D.

2021 *The Turing Test*, in Stanford Encyclopedia of Philosophy (2003).

ORD, T.

2020 *The Precipice: Existential Risk and the Future of Humanity*, Bloomsbury, London; trad. it. *Il precipizio. Il rischio esistenziale e il futuro dell'umanità*, Rizzoli, Milano, 2020.

OWENS, T.

1986 *Vannevar Bush and the Differential Analyzer: The Text and Context of an Early Computer*, in *Technology and Culture* 27(1), pp. 63-95.

PANIMALAR, A. S.; SHREE, V. S.; KATHRINE, V. A.

2017 *The 17 V's of Big Data*. *International Research Journal of Engineering and Technology (IRJET)*, 4(9), 1481-1486. ISSN: 2395-0056.

PARISI, D.

1989 *Intervista sulle reti neurali. Cervello e macchine intelligenti*, Il Mulino, Bologna.

1991 *Il connessionismo oggi*, in Rumelhart, D.E, McClelland, J.L, *PDP. Microstruttura dei processi cognitivi*, Il Mulino, Bologna 1991 trad. it. parz. di Rumelhart, D.E, McClelland, J.L, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1-2, MIT Press.

PARSONS, T.

1951 *The Social System*. Free Press, New York.

1960 *Some Principal Characteristics of Industrial Societies*. In: C. E. Black (a cura di), *The Transformation of Russian Society: Aspects of Social Change since 1861*. Cambridge (MA).

1970 *On Building Social System Theory: A Personal History*. *Daedalus*, 99(4), *The Making of Modern Science: Biographical Studies* (Fall 1970), pp. 826-881.

PASQUALE, F.

2015 *The Black Box Society: The Secret Algorithms That Control Money and Information*, Harvard University Press.

PASQUINELLI, M.

2023 *The Eye of the Master: A Social History of Artificial Intelligence*, Verso Book.

PEIRCE, C. S.

1885 *On the Algebra of Logic: A Contribution to the Philosophy of Notation*. *American Journal of Mathematics*, 7(2), pp. 180-196. <https://doi.org/10.2307/2369451>.

PENNER-WILGER, M., et al.

2007 *The Foundations of Numeracy: Subitizing, Finger Gnosia, and Fine Motor Ability*, in *Proceedings of the Annual Meeting of the Cognitive Science Society*, Vol. 29, No. 29.

PESSA, E.

2001 *Apprendimento automatico e reti neurali*, in Burattini, Cordeschi (eds.), *Intelligenza artificiale*, Carocci, pp. 97-118.

PETERS, B.

2018 *The Computer Never Was a Brain, or the Curious Death and Designs of John von Neumann*, in J. Moser, C.

Vagt (a cura di), *Verhaltensdesign: Technologische und ästhetische Programme der 1960er und 1970er Jahre*, transcript Verlag, Bielefeld, pp. 113–124. <https://doi.org/10.1515/9783839442067-007>

PETRICK, E. R.
2020 *A Historiography of Human–Computer Interaction*, *IEEE Annals of the History of Computing*, 42(4), pp. 8–23. <https://doi.org/10.1109/MAHC.2020.3009080>

PIAS, C. (ed.),
2016 *Cybernetics, The Macy Conferences 1946-1953, The Complete Transactions*, diaphanes.

PICCININI, G.
2003 *Book Review: John von Neumann, The Computer and the Brain, 2nd edition [Book Review]*, *Minds and Machines*, 13(2), pp. 327–332.

2004 *The first computational theory of mind and brain: A close look at McCulloch and Pitts' Logical Calculus of Ideas Immanent in Nervous Activity*, *Synthese* 141(2) 10.1023/B:SYNT.0000043018.52445.3e

2010 *The Mind as Neural Software? Understanding Functionalism, Computationalism, and Computational Functionalism*, *Philosophy and Phenomenological Research*, first published 16 September 2010. doi: 10.1111/j.1933-1592.2010.00356.x. 2025 *Computation in Physical Systems* in *Stanford Encyclopedia of Philosophy*.

2024 *Functionalism, computationalism, and mental contents*, *Canadian Journal of Philosophy*, 34(3), 375–410.

PICKERING, A.
2005 *A Gallery of Monsters: Cybernetics and Self-Organisation, 1940-1970* in *Mechanical Bodies, Computational Minds: Artificial Intelligence from Automata to Cyborgs*.

PINKER, S., PRINCE, A.
1988 *On Language and Connectionism: Analysis of a Parallel Distributed Processing Model of Language Acquisition*, *Cognition*, 28(1), pp. 73–193.

PLATZEK, R. C.; KILBY, J. S.
1964 *Minuteman Integrated Circuits — A Study in Combined Operations*. In *Proceedings of the IEEE*, 52(12), pp. 1669–1678. doi:10.1109/PROC.1964.3466.

PLESNER, A., VONTOBEL, T., WATTENHOFER, R.
2024 *Breaking reCAPTCHA v2*, in *Proceedings of the IEEE International Computer Software and Applications Conference (COMPSAC 2024)*. IEEE. doi: 10.1109/COMPSAC61105.2024.00142.

POGLIANO, C.
2007 *1967: il congresso della Società Filosofica Italiana Cronaca delle quattro giornate pisane*, in *Athenet, Rivista Online dell'Università di Pisa*, n.20, pp.4-8.

POST, E. L.
1936 *Finite Combinatory Processes – Formulation 1*, *Journal of Symbolic Logic*, 1(3), pp. 103–105. <https://doi.org/10.2307/2269031>

PRICE, R.
1982 Price, R. 1992. *Oral History: Claude E. Shannon*. Registrazione di un'intervista orale con Claude E. Shannon, condotta nel 1982. IEEE History Center.

PROUDFOOT, D.
2017 *Child Machines*, in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp. 316-325.

PUGH, E. W.; JOHNSON, L. R.; PALMER, J. H.
1991 *IBM's 360 and Early 370 Systems*. Cambridge (MA), The MIT Press.

- PUTNAM, H. 1975 *Mind Matter and Reality, Philosophical Papers*, vol. II, Cambridge University Press, Cambridge, (MA) (trad. it. *Mente, linguaggio e realtà*, Adelphi, Milano 1987).
- SAPIR E.,
1921 *Language: An Introduction to the Study of Speech*, Harcourt, Brace, New York 1921; tr. it., *Il linguaggio*, Einaudi, Torino 1969.
- SENIOR, A. W., EVANS, R., JUMPER, J., et al.
2020 *Improved Protein Structure Prediction Using Potentials from Deep Learning*, *Nature*, 577, pp. 706–710.
<https://doi.org/10.1038/s41586-019-1923-7>
- SHAPIN, S.
1992 – *Discipline and Bounding: The History and Sociology of Science as Seen through the Externalism–Internalism Debate*, *History of Science*, 30, pp. 333–369.
- SIGNORELLI, A. D.
2023 *Il lungotermismo, l'ideologia del futuro che mette a rischio il presente*, *Il Tascabile*, 17 gennaio 2023.
<https://www.iltascabile.com/scienze/lungotermismo/>
- 2025 *Che cos'è l'accelerazionismo efficace (e chi sono i suoi adepti in Silicon Valley)*, *Wired Italia*, 20 aprile 2025.
<https://www.wired.it/article/accelerazionismo-efficace-silicon-valley/>
- SILVER, D., HUANG, A., MADDISON, C. J., et al.
2016 *Mastering the Game of Go with Deep Neural Networks and Tree Search*, *Nature*, 529, pp. 484–489.
<https://doi.org/10.1038/nature16961>
- 2018 *A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go Through Self-Play*, *Science*, 362(6419), pp. 1140–1144. <https://doi.org/10.1126/science.aar6404>
- SINGER, P.
1972 *Famine, Affluence, and Morality*, *Philosophy & Public Affairs*, 1(1), pp. 229–243; in trad. it. *Etica pratica*, Laterza, Roma–Bari, varie edizioni.
- SMITH, M.
2006 – *How it Began: Bletchley Park Goes to War*, in B. J. COPELAND et al. (a cura di), *Colossus: The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford, pp. 18–35.
- SMOLENSKY, P.
1989 *On the Proper Treatment of Connectionism*, *Behavioral and Brain Sciences*, 11, pp. 1–74; trad. it. *Il connessionismo tra simboli e neuroni*, Marietti, Torino, 1992.
- RAJCHMAN, J.
1980 *Early Research on Computers at RCA* in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.465–470.
- RANDELL, B.
1975 *The Origins of Digital Computers Selected Papers*, 2 ed. Springer-Verlag Berlin Heidelberg New York 1975.
- 1980 *The COLOSSUS*; in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.47–92.
- 2017 *Turing and the Origins of Digital Computers*, in in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.67–76.
- REEDS, J.A.
2015 *Appendix D: Initial Conception of Colossus* in in ed. Reeds, J.A, Diffie, W., Field, J.V., *Breaking Teleprinter Ciphers at Bletchley Park*, John Wiley & Sons, Hoboken (NJ) 2015, pp.535–539.
- RHEINGOLD, H.
1985 *Tools for Thought: The History and Future of Mind-Expanding Technology*, Simon & Schuster, New York.

2000 (2 ed.) *Tools for Thought: The History and Future of Mind-Expanding Technology*, Simon & Schuster, New York.

RICHTMYER, R. D.; VON NEUMANN, J.

1947 *Statistical Methods in Neutron Diffusion*, lettera 11 marzo, in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 751–764.

RIORDAN, M.; HODDESON, L.; HERRING, C.

1999 *The Invention of the Transistor. Reviews of Modern Physics*, 71(2), pp. S336–S345.

ROBERTS, E. H.; YATES, W.

1975 *Altair 8800 / The Most Powerful Minicomputer Project Ever Presented — Can Be Built for Under \$400. Popular Electronics*, January.

ROCHESTER, N., HOLLAND, J. H., HAIBT, L. H., DUDA, W. L.

1956 *Tests on a Cell Assembly Theory of the Action of the Brain, Using a Large Digital Computer, IRE Transactions on Information Theory*, 2(3).

ROGERS, Y.

2012 *HCI Theory: Classical, Modern, and Contemporary*, Morgan & Claypool (Synthesis Lectures on Human-Centered Informatics).

ROJAS, R.

1998 *How to Make Zuse's Z3 a Universal Computer*, *IEEE Annals of the History of Computing*, 20(3), pp. 51–54.

ROSELLI, C., LAPOMARDA, L., DATTERI, E.

2025. *How culture modulates anthropomorphism in Human-Robot Interaction: A review. ACTA PSYCHOLOGICA*, 255(May 2025) [10.1016/j.actpsy.2025.104871].

ROSENBLATT, F.

1958 "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain", in *Psychological Review*, 65(6), pp. 386–408.

1960 *Perceptron Simulation Experiments, Proceedings of the IRE*, 48(3), pp. 301–309.

<https://doi.org/10.1109/JRPROC.1960.287598>

1962 *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington (DC).

ROSENBLUETH, A., WIENER, N.

1950 *Purposeful and Non-Purposeful Behavior*, in *Philosophy of Science*, 17(4), pp. 318–326

ROSENBLUETH, A., WIENER, N., BIGELOW, J.

1943 *Behavior, Purpose and Teleology*, in «Philosophy of Science», vol. 10, n. 1, pp. 18–24; trad. it. *Comportamento, fine e teleologia*, in Wiener, N., *Dio & Golem Spa. Un commento su alcuni punti in cui la cibernetica tocca la religione*, Bollati Boringhieri, Torino 1967.

RUMELHARDT, D., McCLELLAND, J.

1986–1987 *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1–2, MIT Press; trad. it. parz. *PDP. Microstruttura dei processi cognitivi*, Il Mulino, Bologna, 1991.

RUMELHART, D. E., HINTON, G., McCLELLAND, J. L.

1986 *A General Framework for Parallel Distributed Processing*, in RUMELHART, D. E., McClelland, J. L. (a cura di), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*, MIT Press, Cambridge (MA), pp. 45–76; trad. it. *Un quadro generale del processo distribuito in parallelo*, in Rumelhart, D. E., McClelland, J. L., *PDP. Microstruttura dei processi cognitivi*, Il Mulino, Bologna, 1991, pp. 81–116.

- RUMELHART, D. E., HINTON, G. E., WILLIAMS, R. J.
1986a *Learning Representations by Back-Propagating Errors*, *Nature*, 323, pp. 533–536.
- 1986b *Learning Internal Representations by Error Propagation*, in RUMELHART, D. E., MCCLELLAND, J. L. (a cura di), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, MIT Press, Cambridge (MA), pp. 318–362.
- RUSSELL, B. 1902 *Letter to Gottlob Frege* (June 16, 1902). In J. van Heijenoort (Ed.), *From Frege to Gödel: A Source Book in Mathematical Logic, 1879-1931* (pp. 124–125). Harvard University Press.
- RUSSELL, B., WHITEHEAD, A. N.
1910–1913 *Principia Mathematica* (Vols. I–III). Cambridge: Cambridge University Press.
- RUSSELL, S., NORVIG, P.
2021 *Artificial Intelligence: A Modern Approach* (Fourth Edition), Pearson, London.
- RUSSOM, P.
2011 *The Three Vs of Big Data Analytics*. TDWI.
- RUIJTEN, P. A.
2015 *Responses to human-like artificial agents: Effects of user and agent characteristics*.
- RYLE, G.
1949 *The concept of mind*. Barnes & Noble, trad. It. *Il concetto di mente*, Laterza, Bari, 2007.
- SACKS, O.
1985 *The Man Who Mistook His Wife for a Hat and Other Clinical Tales*, Summit Books, New York; trad. it. *L'uomo che scambiò sua moglie per un cappello*, Adelphi, Milano 1986.
- SAGIROGLU, S.; SINANC, D.
2013 *Big Data: A Review. Collaboration Technologies and Systems (CTS)*. 2013 International Conference on Digital Object Identifier, pp. 42–47.
- SAMUEL, A.L.
1959 *Some Studies in Machine Learning Using the Game of Checkers*. IBM Journal of Research and Development, 44, 206–226.
- SARKAR, P.
2000 *A Brief History of Cellular Automata*. *ACM Computing Surveys (CSUR)*, 32(1), pp. 80–107.
<https://doi.org/10.1145/349194.349202>.
- SAVIO, M.
1964 "Speech at the University of California, Berkeley", 2 December.
- SCARANTINO, A.; PICCININI, G.
2010 *Information Without Truth*. *Metaphilosophy*, 41(3), 313–330. <https://doi.org/10.1111/j.1467-9973.2010.01632.x>
- SCHNEIDERMAN, B.
1985 *The Relationship Between COBOL and Computer Science*. *IEEE Annals of the History of Computing*, 7(4), pp. 348–352. <https://doi.org/10.1109/MAHC.1985.10041>.
- SEBESTA, R. W.
2012 *Concepts of Programming Languages* (10^a ed.). Pearson.
- SELBST, A. D., BOYD, D., FRIEDLER, S., VENKATASUBRAMANIAN, S., VERTESI, J.
2018 *Fairness and Abstraction in Sociotechnical Systems*, 2019 ACM Conference on Fairness, Accountability, and Transparency (FAT*), 59–68.

- SELFIDGE, O. G.
1959 *Pandemonium: A Paradigm for Learning*, in *Symposium on the Mechanization of Thought Processes*, National Physical Laboratory, London, pp. 513-526.
- SHANAHAN, M.
2015 *The Technological Singularity*, MIT Press, Cambridge (MA); trad. it. *La rivolta delle macchine. Che cos'è la singolarità tecnologica e quanto presto arriverà*, Luiss University Press, Roma, 2018.
- SHANNON, C. E.
1938 *A Symbolic Analysis of Relay and Switching Circuits*, *Transactions of the American Institute of Electrical Engineers*, Vol. 57, pp. 713-723;
1940 *An Algebra for Theoretical Genetics*, pubblicato in *Collected Papers of Claude Elwood Shannon*, a cura di N. J. A. Sloane e Aaron D. Wyner, IEEE Press, 1993.
1945 *A Mathematical Theory of Cryptography*, Report for the U.S. National Defense Research Committee.
1948 *A Mathematical Theory of Communication*, *The Bell System Technical Journal*, Vol. 27, pp. 379-423, 623-656; trad. it. *La teoria matematica delle comunicazioni*, Mimesis, Milano, 2024
1950 *The Redundancy of English*, in Claude Pias (ed.), *Cybernetics, The Macy Conferences 1946-1953, The Complete Transactions*, diaphanes, pp.248-272.
1950b *Programming a Computer for Playing Chess*, *Philosophical Magazine*, Ser. 7, Vol. 41, No. 314, pp. 256-275.
1953 *Computers and Automata*, in *Collected Papers of Claude Elwood Shannon*, IEEE Press; trad. it. *Calcolatori e automi*, in (a cura di) Somenzi, V, Cordeschi, R., *Filosofia degli automi: Dalla cibernetica all'intelligenza artificiale*, Bollati Boringhieri, Torino 1994, pp.93-107.
1958 *Von Neumann's Contributions to Automata Theory*. *Bulletin of the American Mathematical Society*, 64; in *Collected Papers*, IEEE, 1993, pp. 831-835. doi: 10.1109/9780470544242.ch62.
1979 Interview by Anthony Liversidge. *Omni*, August 1987, 61, 62, 64-66, 110. Reprinted as "Claude Shannon - Interview with Anthony Liversidge," in *Collected Papers of Claude Elwood Shannon*, a cura di N. J. A. Sloane e Aaron D. Wyner, IEEE Press, 1993.
- SHEFFER, H. M.
1913 *A Set of Five Independent Postulates for Boolean Algebras, with Application to Logical Constants*. *Transactions of the American Mathematical Society*, 14(4), pp. 481-488. <https://doi.org/10.2307/1988701>.
- SHERRINGTON, C. S.
1906 *The integrative action of the nervous system*. Yale University Press.
- SHILTON, K., ANDERSON, S.
2016 *Blended, Not Bossy: Ethics Roles, Responsibilities, and Expertise in Design, Interacting with Computers*.
- SIMON, H. A., NEWELL, A.
1958 *Heuristic Problem Solving: The Next Advance in Operations Research*, *Operations Research*, vol. 6, no. 1, 1958, pp. 1-10. JSTOR, <http://www.jstor.org/stable/167397>.
- SINGH, S.
2006 *A Brief History of Cryptography from Caesar to Bletchley Park* in ed. Copeland, B. J, and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.9-17.
- SISMONDO, S.
2010 *An Introduction to Science and Technology Studies* (2nd ed.). Wiley & Sons.
- SLOMAN, A. 1996 *Beyond Turing equivalence* In Peter Millican & Andy Clark, *Machines and Thought: The Legacy of Alan Turing*, Oxford, England: Oxford University Press. pp. 1-179.

SMOLENSKY P.

1989 *On the proper treatment of connectionism*, in Behavioral and Brain Sciences, vol. 11, Cambridge University Press; trad. it., *Il connessionismo tra simboli e neuroni*, Marietti, Torino, 1992.

SONI, J., GOODMAN, R.

2017 *A Mind at Play: How Claude Shannon Invented the Information Age*, Simon & Schuster.

SPREVAK, M.

2017 *Turing's Model of the Mind*, in in Copeland, B.J et al., *The Turing Guide*, Oxford University Press, Oxford 2017, pp.277-286.

STANFORD INSTITUTE FOR HUMAN-CENTERED AI (HAI)

2022 *AI Index Report 2022*, Stanford University.

2023 *AI Index Report 2023*, Stanford University.

2024 *AI Index Report 2024*, Stanford University.

2025 *AI Index Report 2025*, Stanford University.

STIBITZ, G. R.

1942 D Stibitz, Section 2 della Division D, Diary of Chairman, 1 luglio 1942, Boston, Project no. 6, Record Group 227, OSRD, Division 7, General project Files, 1940-1946, General Mathematical Theory of Prediction and Application, MIT Wiener, NDCrc-83, NA-LC, In Galison, 1994, abbr. D.

STINSON, C.

2019 *Explanation and Connectionist Models*, in M. Spreveak, M. Colombo (eds.), *The Routledge Handbook of The Computational Mind*, Routledge, pp. 120-133.

STALLINGS, W.

2008 *Operating Systems: Internal and Design Principles* (6^a ed.). Pearson.

STAMM, E.

1911 *Beitrag zur Algebra der Logik [Contribution to the Algebra of Logic]*. Monatshefte für Mathematik und Physik, 22, pp. 137-149.

STOYAN, H.

1984 *Early LISP History (1956-1959)*. LFP '84: *Proceedings of the 1984 ACM Symposium on LISP and Functional Programming*, pp. 299-310. <https://doi.org/10.1145/800055.802047>.

STRACHEY, C.

1959 *Time Sharing in Large, Fast Computers*. Proceedings of the International Federation for Information Processing (IFIP) Congress 1959.

SUCHMAN, L. A.

1987 *Plans and Situated Actions: The Problem of Human-Machine Communication*, Cambridge University Press, Cambridge.

TAMBURRINI, G.

2020 *Etica delle machine. Dilemmi morali per robotica e intelligenza artificiale*, Carocci, Roma.

TANENBAUM, A. S.; BOS, H.

2022 *Modern Operating Systems* (5^a ed.). Pearson.

TARSKI, A.

1944 *The Semantic Conception of Truth: and the Foundations of Semantics*, Philosophy and Phenomenological Research, Vol. 4, No. 3 (Mar., 1944).

TAYLOR, R.

- 1950 *Purposeful and Non-Purposeful Behavior: A Rejoinder*, in «Philosophy of Science», 17.
- TEGMARK, M.
2018 *Life 3.0: Being Human in the Age of Artificial Intelligence*, Penguin, London; trad. it. *Vita 3.0. Essere umani nell'era dell'intelligenza artificiale*, Rizzoli, Milano, 2018.
- TIME
2023 *The End of Humanity: How Real Is the Risk?*, 12 giugno 2023.
- 2024 *A Roadmap to AI Utopia*, 11 novembre 2024.
- TINNELL, J.
2023 *The Philosopher of Palo Alto: Mark Weiser, Xerox PARC, and the Original Internet of Things*. Chicago, University of Chicago Press.
- THALER, S. L.
1995 *Virtual Input Phenomena within the Death of a Simple Pattern Associator*, *Neural Networks*, 8(1), pp. 55–65. [https://doi.org/10.1016/0893-6080\(94\)00065-T](https://doi.org/10.1016/0893-6080(94)00065-T).
- THALER, R. H., SUNSTEIN, C. R.
2008 *Nudge: Improving decisions about health, wealth, and happiness*, Yale University Press, New Haven CT.
- THIEL, T.
1994 *The Design of the Connection Machine*, *Design Issues*, 10(1), pp. 7–27.
- THIELMANN, T.
2019 *The ENIAC Display: Insignia of a Digital Praxeology* in ed. Haigh, T., *Exploring the Early Digital*, Springer.
- NEW YORK TIMES
1958 *Electronic "Brain" Teaches Itself*, 13 luglio 1958.
- THE TIMES
1950 *Month's Work in a Minute*, in *The Times*, 30 novembre 1950
- TOOLE, B.
1992 *Ada, the Enchantress of Numbers: A Selection from the Letters of Lord Byron's Daughter and Her Description of the First Computer*, Strawberry Press, Londra.
- TORRES, E. P.
2021 *Why Longtermism Is the World's Most Dangerous Secular Credo*, *Aeon*, 14 ottobre 2021.
<https://aeon.co/essays/why-longtermism-is-the-worlds-most-dangerous-secular-credo>
- TUNNE, W.T
2006 *My Work at Bletchley Park* in ed. Copeland, B. J., and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.352-369.
- TURING, A. M.
1936 *On Computable Numbers, with an Application to the Entscheidungsproblem*, in *Proceedings of the London Mathematical Society*, 42, pp. 230-265.
- 1939 *Systems of Logic Based on Ordinals*, in *Proceedings of the London Mathematical Society*, 2^a serie, vol. 45, n. 3, pp. 161–228.
- 1945 *Proposed electronic calculator*, trad. it. In Turing. A, a cura di G. Lolli, *Intelligenza Meccanica*, Bollati Boringhieri Torino 1994, pp.29-62.
- 1947 *Lecture to the London Mathematical Society on 20 February 1947*, trad. it. In Turing. A, a cura di G. Lolli, *Intelligenza Meccanica*, Bollati Boringhieri Torino 1994, pp.63-87.
- 1946 *Lettera a W. Ross Ashby* (19 novembre), in Archivio digitale *Letters to and from W. Ross Ashby*.

- 1948 *Intelligent Machinery*, Report for the National Physical Laboratory, in Copeland, B. J. (ed.), *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*, Oxford University Press, Oxford
- 2004, pp. 411-432; trad. it. *Macchine intelligenti*, in Turing, A. (a cura di G. Lolli), *Intelligenza meccanica*, Bollati Boringhieri, Torino 1994, pp. 88-120.
- 1950 *Computing Machinery and Intelligence*, in *Mind*, 59 (236), ottobre, pp. 433-460; trad. it. *Macchine calcolatrici e intelligenza*, in Turing, A. (a cura di G. Lolli), *Intelligenza meccanica*, Bollati Boringhieri, Torino 1994, pp. 121-158; anche in Turing, A. (a cura di D. Marconi), *Macchine calcolatrici e intelligenza*, Einaudi, Milano 2025.
- 1951 *Programmers' Handbook for the Ferranti Mark I Computer*, University of Manchester, Computing Machine Laboratory, 1951.
- 1952 Gennaio, *Can Automatic Calculating Machines Be Said to Think?* Transcript of BBC radio broadcast, in Copeland, B. J. (ed.), *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*, Oxford University Press, Oxford 2004, p. 495.
- 1952 *The Chemical Basis of Morphogenesis. Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237, pp. 37-72.
- TURING, A., STRACHEY, C., BATES, M. A., & BOWDEN, B. V.
 1953 *Digital Computers Applied to Games*, in Bowden, B. V. (ed.), *Faster Than Thought: A Symposium on Digital Computing Machines*, Pitman, Londra, pp. 286-310.
- TURKLE, S.
 1984 *The Second Self: Computers and the Human Spirit*. Simon & Schuster, New York.
- TURNER, F.
 2006 *From Counterculture to Cyberculture: Stewart Brand, the Whole Earth Network, and the Rise of Digital Utopianism*. University of Chicago Press.
- ULAM, S.
 1958 – John von Neumann 1903-1957, *Bulletin of the American Mathematical Society*, 64(3), pp. 1-49.
- 1976 *The Adventures of a Mathematician*. Scribner, New York.
- 1980 – Von Neumann: *The Interaction of Mathematics and Computing*, in N. METROPOLIS et al. (a cura di), *A History of Computing in the Twentieth Century*, Academic Press, New York, pp. 93-100.
- UMBRELLO, S.
 2021 *AI Winter*, in Klein, M., Frana, P. (eds.), *Encyclopedia of Artificial Intelligence: The Past, Present, and Future of AI*, pp. 7-8.
- 2023 *Oggetti buoni. Per una tecnologia sensibile ai valori*, Fandango Roma.
- VAN DE POEL, I.
 2013 *Translating Values into Design Requirements*, in MICHELFELDER, D., MCCARTHY, N., GOLDBERG, D. (eds.), *Philosophy and Engineering: Reflections on Practice, Principles and Process* (Philosophy of Engineering and Technology, vol. 15), Springer, Dordrecht. https://doi.org/10.1007/978-94-007-7762-0_20
- VAN DEN HOVEN, J.
 2017 *The Design Turn in Applied Ethics*, in *Designing in Ethics*, Cambridge University Press, Cambridge, pp. 11-31. <https://doi.org/10.1017/9780511844317.002>
- VAN WYNSBERGHE, A., ROBBINS, S.
 2013 *Ethicist as Designer: A Pragmatic Approach to Ethics in the Lab*, *Science and Engineering Ethics*, 20, pp. 947-961. <https://doi.org/10.1007/s11948-013-9498-4>

VARELA, F.

1987 *The science and technology of cognition: Emergent directions*, Technical Report for Shell Int.Co. , Londra; trad. it. *Scienza e tecnologia della cognizione. Direzioni emergent*, Hopeful monsters, Firenze 1987.

VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, Ł., POLOSUKHIN, I.
2017 *Attention Is All You Need*, in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS '17)*, Curran Associates Inc., Red Hook (NY), pp. 6000–6010.

VERBEEK, P. P.

2006 *Materializing Morality: Design Ethics and Technological Mediation*, Science, Technology & Human Values.
<https://doi.org/10.1177/0162243905285847>

VON NEUMANN, J.

1925 *Eine Axiomatisierung der Mengenlehre*. *Journal für die reine und angewandte Mathematik*.

1928 *Zur Theorie der Gesellschaftsspiele*. *Mathematische Annalen*, 100(1), pp. 295–320.

1932 *Mathematische Grundlagen der Quantenmechanik*. Springer, Berlin.

1945 *First Draft of a Report on the EDVAC*. Dattiloscritto, 30 giugno 1945, Moore School of Electrical Engineering, University of Pennsylvania.

1946 Lettera a M. Newman, 19 marzo, in Dyson, F. (2012), p. 92, trad. it.

1946b Lettera a N. Wiener, 29 novembre, in *Correspondence, 1946 October–December, Box 5, Folder 72, Norbert Wiener Papers*, Massachusetts Institute of Technology.

1946c Lettera a Norbert Wiener, 26 novembre 1946, in *Norbert Wiener Papers*, Massachusetts Institute of Technology, Institute Archives and Special Collections (consultato online).

1947 *Statistical Methods in Neutron Diffusion*, lettera 11 marzo, in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 751–764 (con R. D. Richtmyer).

1950 *Statistical Treatment of Values of First 2000 Decimal Digits of e and of π Calculated on the ENIAC*. *Mathematical Tables and Other Aids to Computation*, 4, pp. 109–111; in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 765–767 (con N. Metropolis e G. Reitwiesner).

1951 *Various Techniques Used in Connection with Random Digits*. *Journal of Research of the National Bureau of Standards*, Applied Mathematics Series 3, pp. 36–38; in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 769–770.

1951b *The Future of High-Speed Computing*. *Proc. Comp. Sem. I.B.M.*, in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 236–237.

1951c *The General and Logical Theory of Automata*, in Jeffress, L. A. (a cura di), *Cerebral Mechanisms in Behavior: The Hixon Symposium*. John Wiley & Sons, New York, pp. 1–41; trad. it. parz. *La logica degli automi e la loro autoriproduzione*, in Somenzi, V.; Cordeschi, R. (a cura di), *La filosofia degli automi. Origini dell'intelligenza artificiale*, Bollati Boringhieri, Torino 1994, pp. 151–166.

1952 *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*, in Shannon, C. E.; McCarthy, J. (a cura di), *Automata Studies, Annals of Mathematical Studies*, 34, Princeton University Press, Princeton, 1956, pp. 43–98; in *John von Neumann Collected Works*, Vol. V, Pergamon Press, 1963, pp. 236–237.

1958 *The Computer and the Brain*. Yale University Press, New Haven; trad. it. *Il calcolatore e il cervello*, in Somenzi, V.; Cordeschi, R. (a cura di), *La filosofia degli automi. Origini dell'intelligenza artificiale*, Bollati Boringhieri, Torino 1994, pp. 109–150; rist. 2021.

VON NEUMANN, J., MORGENSTERN, O.

1944 *Theory of games and economic behavior*, Princeton University Press.

VON NEUMANN, K.

1957 *Preface*, in von Neumann, J., *The Computer and the Brain*, 1958; trad. it. *Prefazione*, in von Neumann, J., *Computer e Cervello*, Il Saggiatore, Milano 2021.

VOWELS, R.A.

2005 *The Pilot ACE: from project to reality*, in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.223-264.

WANG, H.

1974 *From Mathematics to Philosophy*, Humanities Press, New York.

WATSON J. B.

1913 *Psychology as the Behaviourist Views It*, in "Psychological Review", 20, pp. 158-76, trad. it. *La psicologia così come la vede il comportamentista*, in Id., *Antologia degli scritti*, a cura di P. Meazzini, il Mulino, Bologna 1976, pp. 53-75).

1920 *Is Thinking Merely the Action of Language Mechanism?*, in "British Journal of Psychology", trad. it. *Pensiero e linguaggio*; in Id., *Antologia degli scritti*, il Mulino, Bologna, 1976.

WALDROP, M. M.

2001 *The Dream Machine: J. C. R. Licklider and the Revolution That Made Computing Personal*. New York, Viking Penguin.

WAMBA, S. F.; AKTER, S.; EDWARDS, A.; CHOPIN, G.; GNANZOU, D.

2015 *How "Big Data" Can Make Big Impact: Findings from a Systematic Review and a Longitudinal Case Study*. *International Journal of Production Economics*, 165, 234–246. <https://doi.org/10.1016/j.ijpe.2014.12.031>

WEAVER, W.

1949 *Some Recent Contributions to the Mathematical Theory of Communication*. In Weaver, W., Shannon, C., *A Mathematical Theory of Communication*, 1949, Urbana, IL: The University of Illinois Press; trad. it. in ; trad. it. *La teoria matematica delle comunicazioni*, Mimesis, Milano, 2024.

WEISER, M.

1991 *The Computer of the 21st Century*. *Scientific American*, 265, 94–104. <https://doi.org/10.1038/scientificamerican0991-94>

WEISS, S. M.; INDURKHYA, N.

1998 *Predictive Data Mining: A Practical Guide*. Morgan Kaufmann Publishers, Inc.

WEIR, A.

2024 *Formalism in the Philosophy of Mathematics*, The Stanford Encyclopedia of Philosophy (Fall 2024 Edition).

WEIZENBAUM, J.

1966 *ELIZA — "A Computer Program for the Study of Natural Language Communication between Man and Machine"* in *Communications of the Association for Computing Machinery* 9, pp. 36-45.

WELLS, B.

2006 *The PC User's Guide to Colossus*, in ed. Copeland, B. J. and Others, *Colossus, The Secrets of Bletchley Park's Codebreaking Computers*, Oxford University Press, Oxford 2006, pp.116-140.

WERBOS, P. J.

1974 *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, Tesi di dottorato, Harvard University.

1988 *Backpropagation: Basics and New Developments*, in ARBIB, M. A. (a cura di), *The Handbook of Brain Theory and Neural Networks*, MIT Press, Cambridge (MA), pp. 134–139.

WHEELER, J. A.

1989 *Information, Physics, Quantum: The Search for Links*, report presentato al 3rd International Symposium on Foundations of Quantum Mechanics.

WHITE, M.

2012 *Digital Workplaces: Vision and Reality*. *Business Information Review*, 29(4), 205–214.

WHORF, B. L.

1956 *Language, Thought, and Reality*, MIT Press, Cambridge MA; trad. It. *Linguaggio, Pensiero, Realtà*; Bollati Boringhieri, Torino, 1970.

WIDROW, B., HOFF, M. E.

1960 *Adaptive Switching Circuits*, in ANDERSON, J. A., ROSENFELD, E. (a cura di), *Neurocomputing: Foundations of Research*, MIT Press, Cambridge (MA), pp. 126–134.

WIENER, N.

1929 – *Fourier Analysis and Asymptotic Series*, MIT Press, Cambridge.

1940 Lettera a V. Bush, 20 settembre, in *Correspondence, 1940 September–December*, Box 4, Folder 58, Norbert Wiener Papers, MC-0022, Massachusetts Institute of Technology, Libraries, Department of Distinctive Collections.

1940 Lettera a V. Bush, 21 settembre, in *Correspondence, 1940 September–December*, Box 4, Folder 58, Norbert Wiener Papers, MC-0022, Massachusetts Institute of Technology, Libraries, Department of Distinctive Collections.

1940 – *Memorandum on the Mechanical Solution of Partial Differential Equations*, in P. R. MASANI (a cura di), *Norbert Wiener: Collected Works*, MIT Press, Cambridge.

1942 *Statistical Method of Prediction in Fire Control*, Final Report on Section D-2. Project no. 6, all'attenzione di W. Weaver, Section D-2 NDRC, 1 dicembre 1942, Record Group 227, OSRD, Contractors' Reports, Division 7, NDCrc-83, OSRD Report No. 1863, MIT, NA-LC.

1942b *The Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications*, DIC Contract 6037, MIT 1 febbraio 1942, declassificato, scatola 11, cartella 561°, NWP, pubblicato, 1949, MIT Press, Cambridge MA.

1942c Norbert Wiener a D. I. C 5980 A. A. Directors, «Summary Report for Demonstration», 10 giugno 1942, Record Group 227, Office of Science and Research Development, National Defense Research Committee Contractors' Technical Reports, Division 7, MIT, NDCrc-83, National Archives, Library of Congress, Washington, D. C., in Galison, 1994.

1945 *Operationalism – Old and New* (1945), scatola 11, cartella 570, NWP, pp. 14-15, in Galison, P., 1994 *The Ontology of the Enemy: Norbert Wiener and the Cybernetic Vision*, *Critical Inquiry* 21 (1):228-266 trad. it. in (a cura di) Bianchini, F., Franchi, S., Matteuzzi, M., *Verso un'archeologia dell'intelligenza artificiale*, In "Discipline Filosofiche", XVII I, Quodlibet, Macerata, 2007.

1948 *Cybernetics: Or Control and Communication in the Animal and the Machine*, MIT Press; trad. it. *Cibernetica: Controllo e comunicazione nell'animale e nella macchina*, Il Saggiatore, Milano, 1968.

1948b *Time, communication, and the nervous system*. *Ann N Y Acad Sci*. 1948 Oct 13;50 (Art 4):197-220. doi: 10.1111/j.1749-6632.1948.tb39853.x.

1950 *The Human Use of Human Beings: Cybernetics and Society*, Houghton Mifflin; trad. it. *Introduzione alla cibernetica: Il controllo e la comunicazione nell'uomo e nella macchina*, Comunità, Milano, 1953.

1951 *Homeostasis in the individual and society* in Journal of the Franklin Institute Volume 251, Issue 1, January 1951, pp.65-68, trad. it. in . *Dio & Golem Spa. Un commento su alcuni punti in cui la cibernetica tocca la religione*, Bollati Boringhieri, Torino 1967

1953 *Ex-prodigy: my childhood and youth*. Simon and Schuster.

1956 *I Am a Mathematician: The Later Life of a Prodigy*, Doubleday & Co., Cambridge, MA.

1964 *God and Golem, Inc.: A Comment on Certain Points Where Cybernetics Impinges on Religion*, MIT Press, Cambridge, MA.; trad. it. *Dio & Golem Spa. Un commento su alcuni punti in cui la cibernetica tocca la religione*, Bollati Boringhieri, Torino 1967.

2017 *A Life in Cybernetics: Ex-Prodigy: My Childhood and Youth and I Am a Mathematician: The Later Life of a Prodigy*, MIT press, Cambridge MA.

WILKINSON, J.H

1980 *Turing's Work at the National Physical Laboratory and the Construction of Pilot ACE, DEUCE, and ACE* in in ed. Metropolis et al., *A History of Computing in the Twentieth Century*, Academic Press INC. NY Londra 1980, pp.101-114.

WILLIAMS, F. C.

1975 *Early Computers at Manchester University*, in *Radio and Electronic Engineer*, vol. 45, n. 7, luglio 1975, pp. 327-331

WILLIAMS, F. C.; KILBURN, T.

1948 *Electronic Digital Computers*, in Randell, B., *The Origins of Digital Computers: Selected Papers*, 2^a ed., Springer-Verlag, Berlin-Heidelberg-New York 1975, pp. 387-389.

WILKES, M. V.

2006 *What I Remember of the ENIAC*. *IEEE Annals of the History of Computing*, 28(2), pp. 30-31.

WINNER, L.

1980 *Do Artifacts Have Politics?*, *Daedalus*, 109(1), pp. 121-136.

WINOGRAD, T.

1970 *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language* MIT AI Technical Report 235. 1972 Understanding natural language, *Cognitive Psychology*. 3 (1): 1-191. doi:10.1016/0010-0285(72)90002-3.

WEXELBLAT, R. L.

1981 *History of Programming Languages, Proceedings of the History of Programming Languages Conference*, Los Angeles (CA), 1-3 giugno 1978. Academic Press.

WOLFRAM, S.

1986 *Theory and Applications of Cellular Automata*, World Scientific, Singapore.

2002 *A New Kind of Science*, Wolfram Media, Champaign (IL).

WOODGER, M.

2005 *The ACE Simulator and the Cybernetic Model*, in B. J. Copeland et al., *Alan Turing's Electronic Brain The Struggle to Build the ACE, the World's Fastest Computer*, pp.331-334.

WOOLLEY, T. E.; BAKER, R. E.; MAINI, P. K.

2017 *Turing's Theory of Morphogenesis: Where We Started, Where We Are and Where We Want to Go*, in Cooper, S. B.; Soskova, M. (a cura di), *The Incomputable: Journeys Beyond the Turing Barrier*. Springer, pp. 219-235.

WRONG, D. H.

1996 *Truth, Misinterpretation, or Left-Wing McCarthyism?* *Sociological Forum*, 11(4), pp. 613-621.

XU, Z., JAIN, S., KANKANHALLI, M.

XU, Z., JAIN, S., KANKANHALLI, R.

2024 *Hallucination Is Inevitable: An Innate Limitation of Large Language Models*, arXiv preprint, arXiv:2401.11817.

ZABELL, S.L.

2015 *Statistics at Bletchley Park* in in ed. Reeds, J.A, Diffie, W., Field, J.V., *Breaking Teleprinter Ciphers at Bletchley Park*, John Wiley & Sons, Hoboken (NJ) 2015 pp. lxxv-ci.

ZACH, R.

2003 *Hilbert's Program*, Stanford Encyclopedia of Philosophy.

ZEKRI, O., ODONNAT, A., BENECHÉHAB, A., BLEISTEIN, L., BOULLÉ, N., REDKO, I.

2024 *Large Language Models as Markov Chains*, arXiv preprint. <https://doi.org/10.48550/arXiv.2410.02724>

ZOU, X., XU, S., CHEN, X. ET AL.

2021 *Breaking the von Neumann bottleneck: architecture-level processing-in-memory technology*; in *Sci. China Inf. Sci.* 64, 160404 (2021). <https://doi.org/10.1007/s11432-020-3227-1>

ZUSE, K.

1969 *Rechnender Raum* (trad. ingl. *Calculating Space*, MIT, 1970). Vieweg / Springer.

1984 *Der Computer – Mein Lebenswerk* (autobiografia), Springer, Berlino/Heidelberg. [Prima edizione tedesca; edizione inglese: *The Computer – My Life*, Springer, 1993].

ZWORKYN, V.K.

1951 *Some Prospects of the Field of Electronics*, in "Journal of the Franklin Institute", 251-1 jan.