



ἘΠΈΚΕΙΝΑ

International Journal of Ontology
History and Critics

MICHELE PAVAN

Sulla creatività dei sistemi di calcolo, con una lettura filosofica
del problema della fermata di Alan Turing

EPEKEINA, vol. 16, nn. 1-2 (2023), pp. 1-33
Philosophy of Technology

ISSN: 2281-3209

DOI: 10.7408/epkn.

Published on-line by:

CRF – CENTRO INTERNAZIONALE PER LA RICERCA FILOSOFICA
PALERMO (ITALY)

www.ricercafilosofica.it/epekeina



This work is licensed under a Creative Commons
Attribution-NonCommercial-NoDerivs 3.0 Unported License.

Sulla creatività dei sistemi di calcolo, con una lettura filosofica del problema della fermata di Alan Turing

Michele Pavan

1. La creatività fra psicologia degli albori e intelligenza artificiale. Il modello di Wallas

Il problema di chiarire se una macchina sia capace di azioni o risultati creativi fu esposto per la prima volta nella celebre proposta di Dartmouth, riconosciuta ancora oggi come l'atto di nascita dell'intelligenza artificiale (IA). Nell'incipit del testo del 1955, in cui gli autori propongono che dieci persone per due mesi svolgano una ricerca sull'IA, si dichiara che «lo studio procederà sulla base della congettura per cui, in linea di principio, ogni aspetto dell'apprendimento o una qualsiasi altra caratteristica dell'intelligenza possano essere descritte così precisamente da poter costruire una macchina che le simuli» (McCarthy *et al.* 2006, 413). Uno di questi aspetti, riportato al punto 7, è il “pensiero competente creativo”, definito in quanto distinto dal pensiero competente, ma non creativo, per il fatto di introdurre casualità nel sistema. La congettura per cui l'intelligenza fosse un fenomeno chiarificato al punto da consentire di riprodurlo in una macchina traeva la sua forza da una serie di studi che sul finire del XIX secolo ne formalizzarono gli aspetti come mai era stato fatto in precedenza (Gangemi *et al.* 2003). Per quanto riguarda l'aspetto della creatività,¹ in particolare, alcuni psicologi cominciarono a intenderla come un processo suddivisibile in fasi (Arieti 1986; Trombetta 1989). Un modello che nei primi anni del Novecento riscosse un certo successo fu quello di Wallas, il quale prevedeva una suddivisione del pensiero creativo in 4 fasi: preparazione, incubazione, illuminazione e verifica (2014). Tale modello verrà poi

1. Si assume qui, in linea con una tendenza degli esperti di IA di quel tempo, che la creatività sia uno degli aspetti dell'intelligenza, ma è bene ricordare che diversi studiosi hanno ritenuto come tra i due fenomeni non vi sia una correlazione necessaria. Fra i più celebri sostenitori di questa tesi si ricorda (Guilford 1950). Per un approfondimento e una contestualizzazione storica del lavoro di Guilford vedi (Beaudot 1977).

confermato, seppur con qualche modifica, da numerosi studi e raccolte dati sul modo di operare di poeti, artisti, scienziati e inventori – cioè persone considerate “competenti” e al tempo stesso “creative” (Arieti 1986, 16) – fino a essere ripreso, a partire dalla metà degli anni '50, nelle ricerche sull'IA. In quest'ambito un importante studio su *I processi del pensiero creativo* fu pubblicato qualche anno dopo la proposta di Dartmouth, nel 1962, da Newell, Shaw e Simon (Cantone 2022). Pur essendo in esso esplicito il riferimento a Wallas i tre autori non si limitano qui a una mera applicazione del suo modello in un nuovo ambito, proponendo più ambiziosamente di ridefinire la creatività a partire da una chiarificazione sia contenutistica che metodologica delle teorie esistenti.

Nella letteratura sull'argomento [...] l'attenzione si è concentrata, in particolare, sugli aspetti più spettacolari e misteriosi della creatività: i processi inconsci che si suppone si verificano durante l'“incubazione”, l'immaginazione presente nel pensiero creativo e la sua importanza nel rendere efficace il pensiero, e soprattutto il fenomeno dell'“illuminazione”, l'improvvisa intuizione che rivela la soluzione di un problema, a lungo cercata. [...] Tutti i punti ricordati sono abbastanza interessanti, e rappresentano le parti tipiche di una teoria del pensiero creativo. Nella nostra ricerca sulla creatività, tuttavia, abbiamo avvertito la necessità di avere un'idea più chiara dei requisiti e delle finalità complessive di tale teoria. La nostra proposta è che una teoria del pensiero creativo debba comprendere:

1. La definizione, interamente operativa, del comportamento dei meccanismi (o degli organismi) che, date le opportune condizioni iniziali, pensino effettivamente in modo creativo.
2. Una dimostrazione che i meccanismi che si comportano nel modo definito (da questi programmi) manifestano i fenomeni che abitualmente accompagnano il pensiero creativo (ad esempio, incubazione, illuminazione, formazione e cambiamento di set, e così via).
3. Un insieme di asserzioni, verbali o matematiche, sulle caratteristiche della classe di definizioni (programmi) che comprende i particolari esempi indicati (Cantone 2022, 148).

Anche prendendo per buono il modello di Wallas, una teoria soddisfacente del pensiero creativo dovrebbe insomma rendere conto di

come un sistema x (vivente o non vivente) sia poi capace di agire creativamente in certe condizioni, di come cioè sia *descrivibile in generale* un simile sistema quando in esso si manifestano fenomeni abitualmente associabili al pensiero creativo. In tal senso, la realizzazione di un programma per riprodurre in una macchina i processi descritti da Wallas, oltre che a favorire l'applicazione del suo modello a un ambito nuovo fornirebbe secondo gli autori una *definizione operativa* (punto 1) e al contempo *formale* (punto 3) dei comportamenti abitualmente associabili al pensiero creativo. Tuttavia, lungi dal fornire una *definizione* della creatività, con simili programmi si spiega al limite cosa voglia dire dal punto di vista di una macchina (e al limite – per un' analogia tutta da verificare – di un uomo) agire creativamente *nel senso di Wallas*, il quale non aveva propriamente spiegato che cosa fosse la creatività, limitandosi a descriverne il processo nei termini di un progetto, o problema da risolvere, suggerendo di associarla al modo di operare di persone competenti quali poeti, artisti, scienziati e inventori. D'altra parte, un argomento tipico fra gli esperti IA, come vedremo, è proprio che non sia necessario, e forse nemmeno possibile, disporre di una definizione del genere, e che è se mai attraverso la costruzione di macchine capaci di comportarsi in modi *intuitivamente* intelligenti o creativi che si perverrà forse un giorno a definizioni più precise di questi fenomeni (Veale *et al.* 2019, 3). Gli stessi Newell, Shaw e Simon, sulla scia dell'approccio di Wallas considereranno il pensiero creativo come «un tipo particolare di comportamento di soluzione di problemi» (Cantone 2022, 149), basando questa tesi sull'assunto che «tutte le persone che nel corso della storia hanno promosso i maggiori progressi creativi nella scienza e nella tecnica hanno posseduto capacità molto elevate nella soluzione generale di problemi» (Cantone 2022, 150). La tesi pone qui come ovvio ciò che sarebbe da dimostrare, ovvero che la creatività sia riscontrabile in generale nei calcolatori e, in particolare, in quelli artificiali. Sorvolando su questo punto, gli autori assumono così la correlazione fra creatività e capacità computazionali – dell'uomo e della macchina – per poi individuare in aspetti come la novità o la non convenzionalità dei criteri per distinguere le soluzioni creative dei problemi da quelle generiche. Come vedremo in seguito, aspetti come questi sono effettivamente correlati alla creatività, ma lo sono, aggiungiamo, a prescindere dal fatto che essa si manifesti nella soluzione di un problema. Una procedura di calcolo può anche essere creativa

ma ciò non significa che la creatività sia una caratteristica distintiva, e tanto meno esclusiva del calcolo, così come il fatto che le persone che nella storia hanno promosso i maggiori progressi creativi fossero anche abili nella soluzione di problemi non implica che la creatività sia un'esclusiva dei calcolatori, umani o non umani che siano. Con ciò non si tratterà di sostenere che la creatività non abbia nulla a che fare con la soluzione di problemi ma, in primo luogo, che anche quando la creatività emerge in queste circostanze essa si manifesta in tipi di operazioni e risultati *non essenzialmente* computazionali, e in secondo luogo che tali manifestazioni, descrivibili in termini di novità o non convenzionalità, quando si verificano nel corso di un calcolo implicano in ogni caso una sorta di *aporia operativa*, non essendo *formalmente* possibile il verificarsi di operazioni o risultati creativi – cioè dotati di quegli aspetti manifesti – nel corso di un calcolo.

2. L'approccio soggettivistico all'analogia uomo-macchina

2.1. Il metodo di Turing

L'idea che sia possibile riconoscere in una macchina i fenomeni dell'intelligenza o della creatività senza avere di essi una definizione precisa è un assunto ricorrente negli studi sull'IA. La tesi per cui il problema dell'intelligenza artificiale sia sostanzialmente quello di «costruire una macchina che si comporti in modi che sarebbero considerati intelligenti nel caso di un essere umano» (McCarthy *et al.* 2006, 424) – sulla quale si fonda l'intera proposta di Dartmouth – presuppone un'idea *non chiarita* di che cosa sia l'intelligenza. Tale approccio affonderebbe così le sue radici nel famoso articolo in cui Turing suggerì, forse per primo, di mettere da parte le definizioni generali – nel suo caso quelle di “macchina” e di “pensiero” – per approcciare la questione dal lato pratico, con un esperimento mentale passato poi alla storia come “Test di Turing” (Cantone 2022, 111-137). L'esperimento consiste nell'immaginare un gioco con tre partecipanti: un uomo che pone delle domande e altri due – un uomo e una macchina – chiamati a rispondere. L'interrogante viene chiuso in una stanza, separato dagli altri due. Il suo scopo è determinare, in base alle risposte degli interrogati – pervenutegli attraverso una telescrivente o un intermediario – chi fra i due sia la macchina e chi l'uomo. Scopo della macchina è quello di far

credere all'interrogante di essere un uomo. Scopo dell'uomo è fornire risposte che aiutino l'interrogante a capire la verità. La tesi di Turing è che se l'interrogante non riuscirà a distinguere la macchina dall'uomo allora la macchina pensa, o meglio, che nel caso in cui la macchina svolga in maniera soddisfacente il "gioco dell'imitazione" si avrebbe una prova sufficiente per concludere che essa pensa. Con un approccio retrospettivamente simile ai proponenti di Dartmouth, dunque, Turing inferisce il pensiero nella macchina a partire dalla capacità di essa di riprodurre il comportamento umano. Tale approccio si fonda su almeno due assunti: uno consistente nell'assimilare il pensiero a comportamenti x dell'uomo (assunto oggettivistico), l'altro nel ritenere che un essere pensa allorché un altro essere riconosce nel comportamento del primo la proprietà x (assunto soggettivistico). Considerando che le macchine più adatte a partecipare al gioco dell'imitazione sono per Turing i calcolatori ne consegue che la proprietà del comportamento che qui si suppone sintomatica del pensiero sia la *computazionalità*. Ora, questo modo di approcciare la questione influenzerà, come vedremo meglio nel prossimo capitolo, oltre che i proponenti di Dartmouth, diversi altri esperti IA nel corso del Novecento, rappresentando tutt'ora un importante punto di riferimento negli studi sulle capacità delle macchine di simulare i processi di pensiero dell'uomo. Interessante, da questo punto di vista, è il caso degli studi sulla *creatività*, che pur formando nel loro insieme un ambito nuovo e a sé dell'IA sembrerebbero anch'essi riflettere in qualche misura l'approccio di Turing. In un collettaneo del 2019 – orientato tra le altre cose a definire l'ambito della creatività computazionale (CC), chiarificandone le fasi del recente sviluppo e i progetti a venire – fin dal primo capitolo si afferma che, «date le ovvie difficoltà nel distillare una pura definizione di creatività», i ricercatori CC «ignorano la necessità di definire oggettivamente il fenomeno», e che uno dei loro obiettivi è far sì, se mai, che «le macchine mostrino un livello umano di creatività che anche gli umani percepiranno come creativo» (Veale *et al.* 2019, 3). Riconosciamo qui esplicitamente almeno il secondo assunto, soggettivistico, del metodo di Turing, per il quale una macchina presenta aspetti umani allorché un soggetto li riconosce nel suo comportamento. Che ora l'aspetto in questione sia la creatività, anziché il pensiero in generale, non fa alcuna differenza, poiché in entrambi i casi quel che conta è, per il secondo assunto di Turing, la possibilità stessa che qualcuno riconosca o meno nei modi

della macchina l'aspetto in questione. Per quanto si dica, inoltre, che i ricercatori CC ignorino la necessità di definire oggettivamente la creatività, tale definizione – e con essa il primo assunto di Turing – è qui almeno parzialmente data allorché si assume, come essi fanno, che la proprietà minima del comportamento creativo della macchina sia appunto la computazionalità; va da sé, poi, che a differenza del pensiero in senso lato – che è fatto coincidere, nell'approccio generale di Turing, con il concetto stesso di computazione –, il pensiero creativo dovrà presentare, oltre alla computazionalità, altre e più specifiche proprietà. Ma queste, come vedremo, verranno intese dai ricercatori CC in quanto descrittive di certi *modi di reagire* dell'osservatore – come ad esempio la *sorpresa* – a un comportamento computazionale, motivo per cui non sono da essi incluse, in ultima istanza, nelle proprietà che contribuirebbero a definire *oggettivamente* il fenomeno.

Un altro esempio di applicazione dell'approccio di Turing alla questione CC – e a cui gli studiosi dell'ambito spesso si ispirano – è rappresentato dai lavori di Margaret Boden. In un testo del '90, infatti, divenuto ormai un classico nella letteratura sull'argomento, assumendo che «la creatività umana è qualcosa di misterioso» (2004, 1), Boden afferma che il suo scopo rimane appurare «se i computer potrebbero proporre idee che almeno *sembrano* essere creative» (2004, 7). Tratteremo in modo più approfondito anche questa ipotesi, ma possiamo anticipare che, come nel caso già trattato, suggerendo che un computer – che altro non è che un calcolatore, ovvero una macchina di Turing – sarebbe creativo se qualcuno lo percepisse come tale, anch'essa si fonderebbe sui due assunti del metodo di Turing. Ora, prima di occuparci di tutte queste teorie sarà bene mostrare i limiti che Turing stesso riconosceva al suo metodo. Per quanto riguarda il primo assunto, in *Computing machinery and intelligence* (Cantone 2022) il logico inglese ammetteva che l'equiparazione fra pensiero e computazione fosse una congettura ancora da verificare. Il gioco dell'imitazione può quindi essere letto come un tentativo di verifica di tale congettura. A conferma dell'impostazione congetturale del suo discorso Turing riporta, nel capitolo 6, una serie di «opinioni contrarie a proposito dell'argomento principale». Fra queste vi è l'idea che ciò che fanno le macchine – in una parola: computare – per quanto possa sembrare equiparabile al pensiero umano rimane molto differente da ciò che fa un uomo quando pensa (Cantone 2022, 113). Tuttavia, se la macchina

svolgerà in maniera soddisfacente la sua parte nel gioco dell'imitazione – che in senso più ampio significa: se un calcolatore artificiale svolgerà il suo ruolo in modo tale che un osservatore possa *scambiarlo* per un uomo, cioè un essere pensante – allora l'obiezione per cui il pensiero umano non è computazione rimane irrilevante. Ma in tal modo Turing introduce (anche) nel corso dell'esperimento una seconda regola, nonché un'ulteriore congettura, per la quale il *ricoscimento* da parte di un osservatore dei tratti del pensiero nel comportamento di una macchina sarebbe sufficiente a confermare la prima congettura, cioè l'equiparazione fra pensiero e computazione, e così, più in generale, che le macchine pensano. Tale seconda congettura corrisponde a ciò che abbiamo definito “assunto soggettivistico” ed è anch'esso qui messo alla prova da Turing nella serie delle possibili obiezioni da muovere alla sua tesi. L'obiezione considerata a tal fine è «l'opinione dell'autocoscienza».

Secondo la forma più estrema di questa opinione il solo modo per cui si potrebbe essere sicuri che una macchina pensa è quello di essere la macchina e di sentire se stessi pensare. Allo stesso modo [aggiunge] secondo questa opinione la sola via per sapere che un uomo pensa è quella di essere quell'uomo in particolare. È questo in effetti il punto di vista solipsistico. Può essere il punto di vista migliore cui attenersi sul piano logico, ma rende difficile la comunicazione delle idee. Probabilmente A crederà “A pensa, ma B no”, mentre B crede “B pensa, ma A no”. Invece di discutere in continuazione su questo punto, è normale attenersi alla educata convinzione che ognuno pensi (Cantone 2022, 123-124).

Per difendersi dall'argomento e confermare che anche le macchine pensano vediamo qui come Turing mostri prima quanto sia assurdo, facendo leva su di esso, sostenere che un altro uomo non pensi. Da qui, una volta stabilito che qualsiasi uomo pensa ne segue che basterà osservare che una macchina si comporta come un uomo per convincersi che anch'essa lo fa, e così – in virtù dell'equivalenza emersa, sotto tale aspetto, fra uomo e macchina – che il pensiero coincide in generale con la computazione, fondando in questo modo, di fatto, l'assunto oggettivistico su quello soggettivistico. Ora, da questo quadro è possibile evincere che i limiti del metodo di Turing sono al contempo i suoi punti di forza. Il lato debole del metodo emergerebbe ogni qual volta Turing mostri come si possa criticarlo, rivelando la natura essenzialmente

congetturale dei suoi principi. Tuttavia, le congetture sono proprio, anche, il suo punto di forza, poiché consentirebbero di sorvolare sulle medesime critiche che – per quanto forti, plausibili sul piano logico – si rivelerebbero ininfluenti nella pratica. Ma rimane il fatto che le congetture non sono delle prove. Il test di Turing è importante perché suggerisce che se un uomo scambia una macchina per un altro uomo è probabile che in essa vi sia qualcosa di riconducibile all’umano in generale, ma per provare che sia così bisognerebbe poi ricavare l’“umanità della macchina” dalla macchina stessa più che dalle reazioni di chi la osserva.

In tal senso, se partissimo da una definizione *oggettiva*, anche temporanea, del pensiero, o di altri aspetti da cui si ritiene che sia possibile inferire l’umanità – anziché concludere che questi coincidono con la computazione per il semplice fatto che qualcuno riconosce l’analogia fra il comportamento di un uomo (che presenterebbe per definizione quegli aspetti) e quello di una macchina – si potrebbe all’inverso studiare che cosa *nella macchina*, e dunque *nella computazione*, abbia a che fare con tali aspetti (pensiero, intelligenza, creatività ecc.), allorché il test suggerisca analogie fra il comportamento di quella macchina e il comportamento umano. Tale approccio, che si potrebbe definire *oggettivistico*, non sarebbe dunque in contrasto con i fini pratici del test, anzi, in un certo senso sembrerebbe persino ampliarli, suggerendo la possibilità non solo di cogliere, partendo dalle rispettive definizioni di macchina come “sistema computazionale” e di umano come “sistema pensante, intelligente o creativo”, gli aspetti del secondo nella prima, ma anche, una volta individuato ciò che nella macchina risulta analogo, sotto quegli aspetti, al comportamento umano, definire meglio questi stessi “umani” aspetti *a partire* dallo studio della macchina. Proseguendo in tale direzione, e in conformità con gli obiettivi di questo contributo, nel prossimo capitolo si considererà allora l’analogia uomo-macchina sotto l’aspetto della creatività. In breve, assumendo la definizione per cui *creativo* è “ogni comportamento o risultato che, in rapporto a un dato insieme di regole, risulta *impossibile*”, mostrerò come in base ad alcune implicazioni filosofiche del “teorema della fermata” – noto risultato della logica matematica ottenuto da Turing nel

'36² –, si possa sostenere che esistono operazioni e risultati delle macchine conformi alla definizione, e così che le macchine siano creative in quanto capaci, per certi versi, di fare l'impossibile.

2.2. Il punto di vista sulla creatività

2.2.1 Studi recenti

Abbiamo anticipato come gli studi di CC siano influenzati ancora oggi dal metodo di Turing. Questa influenza, per quanto implichi dei limiti interni, strutturali, in tale ambito di ricerca, presenta in ogni caso degli aspetti positivi. Il fatto di occuparsi di creatività computazionale in termini sperimentali – di realizzare programmi che, pur senza soddisfare un criterio oggettivo di creatività, sarebbero tali da riprodurre comportamenti che nel caso dell'uomo non esiteremmo a definire creativi – potrebbe forse permettere un giorno di chiarire retrospettivamente, a partire dallo studio di programmi più complessi, la definizione vaga, intuitiva, di creatività da cui la ricerca stessa era partita. Tale approccio, che potremmo definire – nel senso buono del termine – scienziata, ha certamente il merito di basare, o di *attendere di basare* le proprie definizioni su fatti osservabili e descrivibili in modo rigoroso. Tuttavia, senza una definizione oggettiva, data in partenza, di creatività, il rischio è quello di scambiare per creativi comportamenti che semplicemente *appaiono* tali a un osservatore, fino al punto che qualunque fenomeno, data l'infinita varietà dei punti di vista, possa essere ritenuto creativo e da qui un possibile candidato per la definizione oggettiva di creatività. È un dato che diversi autori nell'ambito CC condividono oggi tale approccio (Veale *et al.* 2019). Per fare fronte a suoi punti più deboli essi vi includono, oltre all'osservazione in senso lato, alcuni altri criteri, ma sempre di tipo soggettivistico. Graeme Ritchie, per esempio, recupera da Boden (2004) alcuni criteri generici per la definizione di un sistema creativo – “novità”, “qualità/valore”, “tipicità” dei comportamenti o degli output generati (cioè il loro *essere conformi* a quelli noti al sistema, detto anche “accettabilità”) e “sorpresa” che

2. È bene precisare che Turing non ha mai usato – per riferirsi ai risultati ottenuti nell'articolo – le espressioni “teorema della fermata” e “problema della fermata”, introdotte solo successivamente dai suoi interpreti. Anche su questo punto ritorneremo più avanti.

questi generano nell'osservatore (Boden 2004, 168-170) – con l'intento di integrarli in un quadro teorico più ampio e preciso. Seguendo questi criteri, dato un «insieme ispiratore» *I*, cioè l'insieme degli artefatti su cui si basa il comportamento del sistema, una «valutazione della tipicità» *typ* e una «classificazione del valore» *val* di un dato output, diventa possibile considerare ad esempio la novità di quest'ultimo – fattore cruciale per la creatività di un sistema, ma anche generico se riportato da un'osservazione senza criteri – nei termini di «avere un valore basso per *typ* o essere diverso da *I*» (Boden 2004, 170). Ma la bassa tipicità di un output, o la sua differenza da un modello, non sono tali in questo caso dal punto di vista del sistema che lo genera – il quale altro non fa, appunto, che produrre output in base ai modelli di cui dispone – ma da quello del valutatore, che in base a una *sua* idea di tipicità, immaginata a partire dalle fattezze dei modelli forniti al sistema, stabilisce poi, in rapporto a tale idea, se l'output è o non è nuovo, e da qui se il sistema che l'ha prodotto è o non è creativo. D'altra parte lo stesso Ritchie, dopo aver distinto le prospettive di valutazione «descrittiva» (per la descrizione di ciò che accade all'interno dell'attività creativa) e «causale» (per l'identificazione di azioni o precondizioni che, quando si verificano assieme, determinano o costituiscono un evento creativo), riconosce che i suoi criteri definirebbero più precisamente un punto di vista «sintomatico» (Boden 2004, 167-168), il quale, partendo dal presupposto che «la creatività (rigorosa³) è negli occhi di chi la guarda», specifica da una parte «quali fattori osservabili porterebbero un osservatore ad attribuire la creatività al sistema» (Veale *et al.* 2019, 168), e dall'altra le misure che costui «potrebbe applicare per fare affermazioni precise e verificabili su un sistema creativo» (Veale *et al.* 2019, 170). Kazjon Grace e Mary Lou Maher, accogliendo a loro volta che la creatività è negli occhi di chi la guarda, insistono similmente sul fatto che la novità, in quanto aspetto decisivo per l'attribuzione

3. Ritchie distingue appunto due modi di intendere e valutare la creatività: uno vago (*loose*) e uno rigoroso (*strict*). Il modo vago è quello per il quale sarebbero creative tutte quelle attività che, dal punto di vista più ampio di una società, sarebbero considerate di per sé creative: belle arti, scrivere poesie o opere di narrativa, comporre musica ecc. Quello rigoroso, invece, inferisce la creatività di un autore e/o di un atto e/o di un risultato solo e unicamente a partire dal grado di eccellenza che sarà possibile attribuirgli (Veale *et al.* 2019, 161-162).

della creatività, andrebbe misurata a partire dalla «violazione delle aspettative degli osservatori» (Veale *et al.* 2019, 195) piuttosto che dalle differenze oggettive tra gli artefatti. In tal senso, sulla base di un approccio spiccatamente soggettivistico avrebbero ideato un sistema creativo in ambito culinario che, data una ricetta come input è capace, da una parte, di *prevedere* quali ingredienti affini aspettarsi come output, e dall'altra di *suggerire* in base a questa previsione un ingrediente inaspettato (Veale *et al.* 2019, 204-205). Da entrambi i casi si può evincere come i limiti dell'approccio soggettivistico derivino dal fatto che per quanto un output possa apparire atipico o imprevisto a un osservatore qualsiasi (esperto o inesperto che sia), esso rimane perfettamente *possibile* – e a fortiori *tipico e prevedibile* – in base alle regole e ai modelli di cui il sistema dispone per dare output. Tali osservazioni rivelerebbero allora più i difetti computazionali e/o conoscitivi dell'osservatore – relativamente a ciò che *può fare* il sistema osservato – che il potenziale creativo del sistema stesso. Gli autori di (Veale *et al.* 2019) si dimostrano tuttavia consapevoli dei limiti del loro approccio, tanto da riconoscerli espressamente, aggiungendo addirittura che le osservazioni potrebbero rivelarsi utili per identificare eventuali bias umani inerenti all'idea di creatività (3). Ma, ciò nonostante, essi si disinteressano altrettanto deliberatamente di definire la creatività in maniera oggettiva.

2.2.2 Alle radici degli studi recenti sulla creatività computazionale: il caso di Margaret Boden

Questa posizione generale negli studi sulla CC può essere letta alla luce della teoria della creatività di Boden, di cui gli autori di (Veale 2019) tendono ad assumere, in modi che sono più o meno espliciti a seconda dei casi, le premesse di partenza. All'inizio di *The Creative Mind*, condividendo l'idea di Lovelace secondo cui *un computer può fare solo ciò che il suo programma gli permette di fare*⁴ – specificando tuttavia che, pur condividendola, rimarrebbe semplicistica se intesa co-

4. Questa frase di Boden è una versione più contratta e immediata della frase originale di Lovelace: «The Analytical Engine has no pretensions whatever to *originate* anything. It can do *whatever we know how to order it to perform*» (Menabrea-Lovelace 1842).

me argomento a sfavore di qualsiasi nesso fra creatività e computer – la filosofa inglese elenca quattro questioni relative alla creatività, definite *Lovelace-questions* e indicative a nostro avviso, ognuna a suo modo, dei due possibili punti di vista – soggettivistico e oggettivistico – da cui è possibile osservare il fenomeno. Le prime tre questioni indagano, in ordine, «se le idee computazionali possano aiutarci a capire come sia possibile la creatività umana», «se i computer (oggi o in futuro) possano mai fare cose che almeno in apparenza siano creative», «se un computer possa mai sembrare in grado di riconoscere la creatività – ad esempio nelle poesie scritte da poeti umani» (Boden 2004, 16-17). La prima questione è quella che definisce anche, più in generale, la ricerca di Boden, dove lo studio delle dinamiche computazionali dei sistemi artificiali è da sempre funzionale a una maggior comprensione dell'intelligenza e della creatività umane (1987, 2004). La seconda e la terza, sebbene secondarie, per ammissione stessa dell'autrice sono cruciali allorché «gettano luce sulla prima» (Boden 2004, 17). Nell'ottica del presente contributo, inoltre, lo sono in quanto rivelano la prospettiva soggettivistica entro cui Boden considera la creatività dei sistemi artificiali. L'indagine della seconda questione, infatti, presuppone che si consideri questi sistemi dalla prospettiva di un soggetto qualsiasi (che potrebbe essere tanto un individuo – esperto o inesperto non cambia – quanto una società) tenendo conto di ciò che lo spingerebbe ad attribuire loro capacità creative (nel caso di Boden: la novità, il valore e l'imprevedibilità degli output generati), mentre l'indagine della terza presuppone che si consideri, dal medesimo punto di vista, se tali sistemi siano capaci di riconoscere in quanto creativi quegli output che i loro osservatori riconoscerebbero come tali. Ma veniamo ora, invece, alla quarta questione, ovvero stabilire «se i computer stessi possano mai essere realmente creativi (invece di limitarsi a produrre prestazioni apparentemente creative la cui originalità è interamente dovuta al programmatore umano)». È significativo che Boden riveli fin da subito di non volerla trattare.⁵ Essa, infatti, richiederebbe un'indagine squisitamente oggettivistica del fenomeno in questione. E ciò almeno

5. Per la precisione Boden rimanda all'ultimo capitolo la questione, ma anche in quel frangente non mostra come sia possibile rispondere "sì" alla questione ma solo come sia possibile confutare gli argomenti di chi risponderebbe "no", lasciandola così aperta.

in due sensi: da una parte perché, ed è il senso a cui allude Boden, essa invita a considerare se l'*oggetto* osservato sia l'*autore reale* delle operazioni creative che esegue e dall'altra perché invita a considerare se tali stesse operazioni siano realmente creative (e non solo *percepite come tali* da un soggetto qualsiasi). Ora, sebbene Boden non spieghi i motivi della resistenza a indagare il fenomeno da questo punto di vista, limitandosi a dire che la quarta questione non verrà affrontata nel libro, mi sembra possibile quanto meno suggerirne due: uno di carattere storico-metodologico e l'altro di carattere teoretico-speculativo.

Partendo dal primo, si potrebbe sostenere che Boden non consideri il punto di vista oggettivistico poiché richiederebbe da una parte di stabilire una volta per tutte una definizione oggettiva di creatività e dall'altra di non affidare più la valutazione della creatività effettiva di un sistema artificiale all'osservazione di un soggetto, il che significherebbe in entrambi i casi venir meno a quell'approccio pragmatico e sperimentale dei suoi predecessori (da Turing a Newell, Shaw e Simon) a cui sembra lei stessa allinearsi (2004, 10).⁶ Per quanto riguarda il secondo, invece, si potrebbe sostenere che Boden non consideri questo punto di vista in quanto, se si stabilisce, come del resto essa fa, che la tesi di Lovelace è corretta, cioè che *un computer può fare solo ciò che il suo programma gli permette di fare*, ne consegue che esso non possa essere *realmente* creativo e che non avrebbe dunque alcun senso indagare la creatività da un punto di vista oggettivistico. Ed è proprio per questo motivo che, al fine di mantenere aperta la possibilità che le macchine siano comunque creative, senza abbandonare la tesi di Lovelace, Boden sposterà, come già Turing fece anni prima,⁷ il luogo in cui cercare la creatività dell'*oggetto* osservato dall'*oggetto* medesimo al suo osservatore. Tale approccio la porterà a definire tre differenti forme di creatività (combinatoria, esplorativa e trasformativa) (Boden

6. In questo passaggio di *The Creative Mind* è avvertibile la prudenza di Boden nel definire oggettivamente la creatività, in quanto, essendo un fenomeno strettamente correlato con la scala di valori di chi lo valuta rimarrebbe non isolabile da un punto di vista particolare.

7. Turing ragiona, come si è visto, sulla caratteristica umana del pensiero; ma parla anche di creatività, nello stesso articolo, proprio a partire dalla tesi di Lovelace, facendo vedere che una macchina può, nonostante Lovelace, generare risultati in grado di sorprendere il suo programmatore, e cioè risultati che in un certo qual modo non rispecchierebbero gli ordini impartiti (Cantone 2002, 128).

2004, 3)), nessuna delle quali implicante l'esecuzione, da parte di una macchina, di operazioni realmente creative, cioè realmente diverse da quelle che il programmatore le ha ordinato di eseguire. Persino nel caso della creatività trasformativa – che per Boden è la più radicale –, per quanto una macchina possa infatti modificare il proprio stesso spazio concettuale, consentendo operazioni che prima di ciò non sarebbero state possibili, rimane il fatto che essa, per la tesi di Lovelace, non ha altro modo di eseguire tale trasformazione se non eseguendo ordini iscritti in questo spazio, da cui consegue che anche le operazioni che prima della trasformazione si è creduto non fossero possibili in realtà lo erano considerando il tipo di trasformazioni di se stessa che la macchina poteva compiere in base a quegli stessi ordini, e che la percezione dell'incapacità della macchina di eseguire quelle operazioni dipendeva allora più dai limiti del programmatore nel comprendere le possibilità reali della macchina – cioè del programma, o insiemi di ordini, che lui stesso ha creato – che da quelli della macchina chiamata a eseguirle. C'è da dire che su questo punto Boden rimane un po' ambigua. Parlando di creatività trasformativa, in diversi punti essa afferma che le trasformazioni che un sistema applica a se stesso gli rendono a partire da quel momento possibili una serie di operazioni che non potevano esserlo prima. D'altra parte, però, sappiamo che Boden ritiene anche corretta la tesi di Lovelace, e così che tutto ciò che una macchina fa è necessariamente *sempre stato possibile* in base alle sue regole. Questo potrebbe forse spiegare il motivo per cui Boden parla, in definitiva – anche nei casi di creatività trasformativa – di sistemi che sembrano essere creativi ma che non (sappiamo se) lo sono *realmente*. E tuttavia, proprio in virtù di tale ambiguità di fondo si potrebbe sospettare che Boden non sia mai stata del tutto persuasa dalla tesi di Lovelace secondo cui *un computer può fare solo ciò che il suo programma gli permette di fare*. Il modo con cui la filosofa inglese descrive la creatività trasformativa lascia intendere com'ella potrebbe aver colto, fra le pieghe di questa idea, delle evidenze per le quali bisognerebbe concludere, al contrario, che i computer fanno cose che i loro programmi non gli permetterebbero di fare, salvo poi decidere, più o meno consapevolmente, di proseguire le sue ricerche in sicurezza, mantenendosi nei limiti tracciati da Lovelace e lasciando impensata la questione della *creatività reale* dei sistemi artificiali.

Ma il caso di Boden è allora interessante per due motivi antitetici. Da una parte perché essa farebbe da spartiacque fra la generazione di studiosi che ha pensato i problemi del pensiero e dell'intelligenza dei sistemi artificiali secondo il paradigma della computazione e quella che, anche grazie al suo contributo, ha pensato sotto lo stesso paradigma la creatività, e dall'altra in quanto nel mezzo di tale passaggio essa mostrerebbe anche, senza tuttavia approfondirli, i limiti stessi di tale paradigma, per il quale appunto si sosterebbe che – per quanto possa essere strana – qualsiasi caratteristica (pensiero, intelligenza, creatività) ritenuta attribuibile a un sistema che è – fra le altre cose (e il punto è proprio capire cosa tali “altre cose” siano) – capace di calcolare, gli è a tutti gli effetti attribuibile se e solo se *qualificante* operazioni *conformi* a regole o istruzioni di cui il sistema dispone. Parlando di creatività trasformativa, infatti, abbiamo visto come Boden dicesse di riferirsi a una situazione in cui il sistema esegue delle operazioni che fino a prima della trasformazione *non* sarebbero state *possibili*, cioè che non sarebbero state quanto meno *conformi* all'insieme delle regole di cui esso disponeva. Per capire ora come in tale idea venga scoperta, ma al contempo persa di vista, la questione della *creatività reale* (o *radicale*) dei sistemi artificiali, c'è da chiedersi quanto segue: in che senso si può dire che fino a prima che la macchina trasformasse se stessa determinate operazioni non erano possibili se è pur vero che questa trasformazione (e dunque tutto ciò che grazie a essa diventa possibile, comprese quelle operazioni) dev'esserlo sempre stata? Prima di rispondere vediamo di chiarire l'assunto interno alla domanda. Che una simile trasformazione debba essere sempre stata possibile segue direttamente dal fatto che se la macchina la esegue significa che essa è stata programmata per eseguirla e che dunque tale possibilità di esecuzione è sempre esistita per quanto mai realizzata. L'alternativa sarebbe dire che anche questa trasformazione non sarebbe stata di per sé possibile se non in virtù di un'ulteriore trasformazione, ma più si continua a negare in questo modo che le trasformazioni che una macchina applica a se stessa non sono possibili in base alle regole di cui la macchina dispone – ma solo grazie ad altre trasformazioni che le rendono possibili – più diventa evidente che a un certo punto, da qualche parte, devono esservi state nella macchina delle regole che hanno reso possibile almeno una prima trasformazione e da qui, come per un effetto domino, tutte le trasformazioni successive, compresa

quella che renderà possibile, alla fine della serie, le operazioni ritenute fino a quel momento impossibili. Chiarito questo punto possiamo ora rispondere alla domanda. Possiamo dire, in particolare, che se è vero che la trasformazione che ha reso possibile le operazioni fino a quel momento impossibili è sempre stata possibile in base alle regole di cui il sistema disponeva ne consegue che lo erano anche quelle operazioni malgrado potessero apparire impossibili a un osservatore incapace di prevederle sulla base delle trasformazioni che il sistema avrebbe potuto applicare a se stesso.

Fin qui, dunque, nulla che possa *davvero* scuotere la tesi di Lovelace. Le modifiche che la macchina può apportare al proprio spazio concettuale, proprio in quanto possibili non tolgono, anzi confermano, che nella macchina tutto procede secondo le sue regole. Il punto centrale è che la creatività trasformativa, per come è concepita da Boden, non sembra sufficiente per parlare di *creatività reale*. Affinché un sistema possa essere realmente creativo deve mostrare di eseguire non tanto operazioni rese possibili da delle trasformazioni a loro volta possibili ma operazioni *realmente impossibili*. Deve poter fare ciò che il suo programma non gli permetterebbe di fare. Da questo punto di vista, allora, non avrà alcun senso distinguere un'operazione da una trasformazione che la rende possibile. Quello che conta per l'attribuzione della creatività reale a un sistema è che un'operazione da esso eseguita sia impossibile in base alle regole di cui dispone (cosa intravista e tuttavia non approfondita da Boden, che finirà per scambiare l'*impossibilità reale* con la *possibilità imprevedibile* o *impossibilità apparente* di un'operazione). Se questo criterio è soddisfatto va da sé poi che quell'operazione sia anche *trasformativa*; non nel senso, però, che è programmata per far sì che un'operazione impossibile, o meglio inconsueta o imprevedibile per un osservatore, si realizzi, ma perché non essendo propriamente possibile allora la sua effettiva esecuzione deve aver comportato in qualche misura l'istituzione di una regola nuova. Torneremo più avanti su questo punto. Prima di arrivarci sarà essenziale chiarire, nel prossimo capitolo, in che termini la nozione di creatività sia indissolubilmente legata a quella di impossibilità. Solo allora potremo mostrare, facendo leva su alcune implicazioni del "problema della fermata", che le macchine eseguono necessariamente delle operazioni impossibili e che dunque, per via di ciò, si potrebbe dire che siano capaci di creatività reale, nonché di creare le proprie regole.

3. La creatività reale. Per un approccio oggettivistico alla questione della creatività computazionale

3.1. Dal programmatore al programma

3.1.1 Questioni di metodo e definizioni generali

Nei capitoli precedenti abbiamo visto come nel sondare le potenzialità dei sistemi artificiali sia i ricercatori di IA che di CC non amino partire da definizioni generali di aspetti umani quali il pensiero, l'intelligenza e la creatività, confidando che queste potranno essere, in futuro, direttamente illuminate da una ricerca condotta sul campo, preliminarmente fondata su assunti ragionevoli. Per quanto riguarda il presente contributo, lungi dal contestare l'importanza di un approccio sperimentale, si ritiene altrettanto importante che i così detti assunti ragionevoli, e in particolare le definizioni preliminarmente accettate di creatività, si riferiscano direttamente agli oggetti (in questo caso i sistemi artificiali) a cui la ricerca è rivolta, piuttosto che agli effetti (meraviglia, sorpresa, incredulità) che la ricerca può generare in chi la conduce – sia esso un soggetto concreto, in carne e ossa, o immaginario (una società intesa come insieme di idee condivise su che cosa sia la creatività). Il motivo è presto detto. Si crede qui fermamente che la ricerca negli ambiti IA e CC possa davvero illuminare questi fenomeni – così come lo hanno sempre fatto, a loro modo, la psicoanalisi, la psicologia, l'antropologia e la sociologia. Ma proprio in quanto si crede ciò, si dovrà anche ritenere che il miglior modo con cui tali ricerche potranno mai contribuire alla causa sarà partendo dai loro stessi oggetti, per poi individuare da qui aspetti, operazioni o risultati conformi alla definizione generale, extra-disciplinare, del fenomeno indagato, piuttosto che indagare lo stesso fenomeno a partire dagli *effetti* che il proprio oggetto può generare in soggetti come l'uomo o la società. Quest'ultima indagine, infatti, potrà anche essere utile ai fini di un'inchiesta psico-sociologica delle aspettative – e degli eventuali *bias* cognitivi a esse associate – relativamente all'idea di creatività o a quella delle reali potenzialità di un sistema artificiale, ma non dice nulla attorno a queste stesse potenzialità e non contribuisce quindi ad un aumento del livello di conoscenze tecnico-specifiche in IA e CC. Per condurre una ricerca sui sistemi artificiali, mantenendo il focus sulla questione della creatività, bisognerà dunque avere, innanzitutto,

una definizione che sia il più possibile generale e oggettiva di questa proprietà. Una volta stabilita, si tratterà poi di verificare se vi siano aspetti, operazioni o risultati delle macchine, o di qualche macchina in particolare, conformi alla definizione. Tale verifica può avvenire in due modi, a seconda che il tipo di ricerca che si è scelto di svolgere sia empirico-induttivo o logico-deduttivo. Nel primo caso si procederà prendendo in considerazione particolari sistemi – come potrebbe essere un programma per il gioco degli scacchi – per valutare la conformità delle sue operazioni – in questo caso delle sue mosse – alla definizione di creatività. Nel secondo si tratterà invece di valutare se le macchine di Turing – ovvero gli attuali computer – possano in generale eseguire operazioni conformi alla definizione data.

Quello che si cercherà di fare da qui in avanti sarà proporre un inizio di possibile ricerca del secondo tipo. Procediamo subito con l'individuazione di una definizione di creatività che sia il più possibile generale e oggettiva. Un buon punto di partenza potrebbe essere identificare il *tipo* di azioni a cui sarebbe possibile attribuire questa proprietà. Rimanendo il più possibile generali e oggettivi va da sé che dovremmo definire tali azioni "creazioni". Che cos'è, dunque, una creazione? Qualcosa di nuovo, certo. Ma a questo punto – ed è qui che iniziano i problemi – bisognerà anche chiedersi: è possibile che le macchine creino *in tal senso*, che producano cioè dei risultati *nuovi*? Se esse agiscono sempre, per la tesi di Lovelace, sulla base delle regole di cui dispongono, la risposta alla domanda non può che essere negativa, poiché quelle regole produrranno sempre, relativamente a un dato problema, gli stessi risultati, o quanto meno risultati dello stesso genere di quelli già prodotti, in passato, in base a quelle regole. Tuttavia, anziché concludere, alla luce di ciò, che le macchine non sono creative, potremmo invece includere la nozione di *impossibilità* nella nostra definizione, aprendo da qui, però, la questione di capire in che modo le macchine, pur potendo agire solamente in base alle proprie regole, possano produrre risultati *formalmente* impossibili ed essere creative nel senso stabilito. Ricaviamo dunque dalla risposta negativa alla domanda la seguente *definizione positiva, generale e oggettiva* di creatività computazionale: "creativo è qualsiasi atto o prodotto che,

in rapporto alle regole del sistema considerato, risulta impossibile”.⁸ Proponiamo ora di considerare alcune implicazioni del teorema (o del problema) della fermata come argomenti utili a provare che le macchine sono capaci di atti o risultati conformi alla definizione.

3.1.2 Introduzione storica al teorema della fermata

Il teorema della fermata⁹ fu dimostrato e presentato per la prima volta da Turing nel 1936 in un articolo intitolato *On computable numbers, with an application to the Entscheidungsproblem*.¹⁰ L’obiettivo principale di Turing era fornire, come recita il sottotitolo, una risposta al problema matematico della decisione, presentato da David Hilbert in un primo momento nel 1900, a Parigi, nel corso del Secondo Congresso Internazionale dei Matematici, per essere poi riportato alla ribalta, nel 1928, sempre da Hilbert, nello stesso congresso tenuto a Bologna, in quanto dalla sua prima formulazione nessun matematico lo avevo ancora risolto. Il problema consisteva nel capire se vi fosse un «metodo sistematico per dire, di ogni proposizione matematica, se la proposizione è dimostrabile o meno nel sistema» (Copeland 2004, 47). Nel suo articolo Turing provò, in buona sostanza, che questo metodo non esiste, o meglio che non esiste un modo per decidere se, dato un problema matematico a un qualsiasi sistema di calcolo, il sistema lo risolverà o meno. Già nel ’31 Kurt Gödel provò che dall’interno di qualsiasi sistema formale coerente e abbastanza potente da contenere l’aritmetica è impossibile dimostrare, ad esempio, una proposizione che dice di se stessa di non essere dimostrabile nel sistema. Infatti, se si dimostra la proposizione g “ g non è dimostrabile nel sistema S ” si ottiene che è vero che g non è dimostrabile in S e così una contraddizione. A questa idea,

8. Una definizione simile è stata fornita anche da Boden senza essere tuttavia sviluppata (Dartnall *et al.* 1994, 6; Boden 2004, 6). L’intento stesso di Boden in *The Creative Mind* è, fra gli altri, quello di calmierare in un certo qual modo il paradosso che la definizione implica, anziché assumerlo e svilupparne le implicazioni (13).

9. Per un approfondimento su come Turing arrivò ai risultati presentati nel ’36 si veda (Hodges 2012; Copeland 2004)

10. È in realtà Martin Davis il primo a nominare la dimostrazione di indecidibilità di Turing con l’espressione “problema della fermata” (halting problem) (Davis 1958). Per una ricostruzione storica delle vicende che portarono a definire in questi termini il problema esposto da Turing nel ’36 (Lucas 2021).

da cui deriva che un sistema non è in grado di dimostrare *tutte* le sue verità – e che dunque, per rimanere *coerente*, dovrà essere *incompleto* – Turing aggiunse che non è nemmeno possibile *decidere* se lo stesso sistema dimostrerà *una* qualsiasi delle sue verità.

Ora, la connessione esistente fra questo risultato, valido per la logica-matematica in generale, e la particolarità del teorema della fermata – da cui emerge, fra le altre cose, la validità del risultato per l'informatica – la si può capire alla luce del modo con cui Turing affrontò l'*Entscheidungsproblem*. Dal momento che il metodo richiesto da Hilbert per decidere la dimostrabilità di una proposizione matematica in un sistema doveva essere a sua volta un sistema di calcolo, Turing immaginò una macchina che potesse riprodurre le operazioni compiute da un calcolatore umano. Per fare ciò era necessario soddisfare una serie di condizioni. Innanzitutto, la macchina doveva poter operare con un certo insieme di simboli su un nastro diviso in celle (l'equivalente di una riga di un foglio a quadretti). A tal fine sarebbe stato necessario fornire alla macchina un *alfabeto* – composto dai simboli che le si vorrà far utilizzare – e una *testina* per operare sul nastro leggendo, scrivendo o cancellando tali simboli. Ciascuna cella del nastro avrebbe dovuto contenere un solo simbolo alla volta e la testina si sarebbe dovuta collocare, in ciascuna fase del calcolo, a livello di una singola cella, in modo che per svolgere le successive operazioni su altre celle – come scrivere un risultato dopo una lettura, o leggere un nuovo simbolo dopo averne scritto uno – avrebbe dovuto spostarsi o a destra o a sinistra, salvo trovarsi nella condizione di dover cancellare il simbolo della cella in cui si trova per sostituirlo con un altro. Oltre a ciò, dal momento che l'esecuzione di un calcolo richiede di tenere presenti le operazioni già eseguite e i simboli osservati in precedenza, bisognerà far sì che la macchina possa assumere uno alla volta – uno per ogni nuova operazione svolta durante il calcolo – un certo numero di *stati interni*, corrispondenti agli *stati mentali* che può assumere di fatto un uomo allorché pensa un nuovo contenuto oltre a quelli che sta tenendo a mente mentre calcola. In questo modo si potrà dire che la *configurazione* data da uno stato interno e un nuovo simbolo determinerà il possibile comportamento successivo della macchina nel corso del

calcolo. Soddisfatte tutte queste condizioni,¹¹ dunque, le macchine di Turing (d'ora in avanti MT) diventerebbero capaci di svolgere qualsiasi calcolo. Non solo: se programmate in modo adeguato lo svolgerebbero meccanicamente e senza commettere errori, così come una calcolatrice ben programmata risponderà "4" se si digita sulla sua tastiera (equivalente del nastro pensato da Turing) il problema "2+2=". A questo punto non rimaneva che chiedersi: può una MT risolvere il problema di Hilbert? A rigore, essendo una MT un sistema di calcolo, ed essendo che Hilbert chiedeva di fornire un metodo per decidere, di ogni proposizione matematica, se la proposizione sarebbe stata dimostrabile o meno in un sistema di questo tipo, ne derivava che per soddisfare la richiesta una MT¹ doveva essere in grado di calcolare se un'altra MT² avrebbe dimostrato o meno una proposizione P: se datole cioè in ingresso un qualsiasi input – per esprimersi in termini più informatici – avrebbe terminato o meno il calcolo. Per mettere MT¹ nelle condizioni di farlo Turing mostrò, in primo luogo, come l'insieme delle informazioni contenute in MT² – relative alle operazioni che essa può svolgere allorché dotata delle caratteristiche di cui sopra – una volta scritte avrebbero potuto formare una «tabella di comportamento» di dimensioni finite, tale da esprimere, fra le altre cose, il comportamento di MT² con input P. Ciò significava che era possibile inserire la tabella di MT² in un elenco interno a MT¹ – opportunamente tradotta, ovviamente, con l'alfabeto di cui la macchina dispone –, attribuirle un «numero di descrizione» (ad esempio 1) tale da poterla distinguere da altre tabelle inserite nell'elenco e così far sì che riportandone il numero sul suo nastro MT¹ potesse decodificarlo nella tabella corrispondente a MT² e infine eseguirlo. In altre parole, con tale sistema era possibile far sì che MT¹ simulasse il comportamento di MT² e di qualsiasi altra macchina traducibile nei simboli del suo alfabeto. Era la prova che le macchine di Turing fossero *macchine universali* (d'ora in avanti MTU). Ma a questo punto ecco il colpo di scena. Una volta stabilito che una MT poteva simulare qualsiasi altra MT diventava possibile chiedersi: la MTU così definita può decidere se, dando alla MT che essa è in grado di simulare un qualsiasi input P, tale MT terminerà il suo calcolo o meno?

11. Per un elenco completo e al tempo stesso sintetico delle caratteristiche di una MT si veda (Frixione *et al.* 2011, 51-53).

La risposta era no. Poteva effettivamente svolgere il calcolo della MT considerata, ma non poteva stabilire in anticipo se tale MT l'avrebbe risolto. Per farlo, infatti, avrebbe dovuto essa stessa risolverlo al suo posto. Come avrebbe potuto, del resto, prima di ciò? Com'è possibile, in generale, stabilire se un insieme di regole – cosa che del resto è una MT – sono adatte a risolvere un problema, ovvero ad arrivare a una soluzione in un numero finito di passi, se non già tentando di risolverlo applicando quelle regole? Avventurarsi in questo tentativo non escluderebbe, ovviamente, che sia possibile risolvere il problema, ma precisamente che sia possibile decidere, prima di risolverlo, se sia risolvibile o meno, anche perché se si suppone che prima di arrivare alla soluzione si avrà capito, come in anticipo, con quanti passi si risolverà il problema, quello definirà esattamente l'istante in cui lo si avrà risolto. Turing darà una dimostrazione formale dell'irrisolvibilità del problema della fermata, provando conseguentemente che il problema di Hilbert è indecidibile, cioè che non esiste un metodo sistematico per decidere se è possibile risolvere una proposizione matematica dall'interno di un qualsiasi sistema programmato per farlo.

3.1.3 Implicazioni filosofiche del problema della fermata

Ora, ai fini di questo contributo non sarà necessario ripercorrere la dimostrazione. Dobbiamo capire, invece, quali siano le implicazioni del problema della fermata tali da dimostrare che le operazioni svolte da qualsiasi MT – cioè da qualsiasi computer – siano conformi alla definizione di creatività che abbiamo dato. Consideriamo a tal fine un caso particolare. Tra le varie operazioni che una MT può svolgere Turing mostra che vi è anche quella di simulare se stessa – dal momento che anche le sue informazioni sono di fatto convertibili in una tabella che essa può codificare ed eseguire (tale possibilità è esaminata da Turing nella sezione 8 del suo articolo). Ciò significa che, per gli stessi motivi di cui sopra, non solo una particolare MT non può calcolare se un'altra MT terminerà o meno un calcolo... non lo può calcolare nemmeno di se stessa. Ma dal momento che qualsiasi impossibilità di una MT è il corrispettivo di una regola di cui la MT non dispone, il fatto che una particolare MT non possa calcolare se è in grado di risolvere un calcolo rivela che essa *non dispone delle regole per risolvere quel calcolo con le regole di cui dispone*, e che dunque se alla fine lo

risolverà con queste regole eseguirà un'operazione almeno *in parte* impossibile.

Esistono vari modi per esprimere questo problema. Proviamo allora a prenderne in considerazione un altro. Bisogna chiarire innanzitutto che non stiamo affermando che una MT non sia programmata per risolvere problemi. Essa, infatti, dispone di una serie di regole per farlo. Tuttavia, essa non dispone delle regole che le consentono di calcolare se le regole di cui dispone siano adeguate per farlo, ed è questo, in effetti, il significato profondo di “non sa dire se terminerà/risolverà o meno un calcolo con le regole di cui dispone”. Se avesse quelle regole, infatti, essa potrebbe risolvere il problema della fermata per i calcoli da fare con le regole di cui dispone, salvo che il problema si ripresenterebbe di nuovo per le regole con le quali è in grado di stabilire se le regole di cui dispone la porteranno alla risoluzione/terminazione del calcolo. E così via all'infinito. Ciò confermerebbe, di nuovo, che il problema della fermata non è risolvibile, perché non sapremo mai in anticipo se, a un certo punto, troveremo una regola per la quale tutte le regole di cui la MT dispone – pensate per risolvere un calcolo qualsiasi – la porteranno alla risoluzione/terminazione del calcolo. Questo perché, come si è detto, se e quando troveremo quest'ultima regola fondativa di tutte le altre a quel punto avremo *già* risolto il calcolo, ma di nuovo senza una regola per stabilire se la regola fondativa (e così tutte le altre) ci porterà o meno alla risoluzione/terminazione del calcolo stesso, pur avendolo in effetti risolto. In questo caso si potrà quindi dire che il calcolo è sì *terminato in avanti*, ma che *non* è ancora *terminato all'indietro*, cioè che il calcolo è *terminato* e al tempo stesso *non è terminato*, e che dunque, di nuovo, è indecidibile il problema se, data a una MT una qualsiasi proposizione P, MT terminerà o meno il calcolo. Generalizzando questi risultati è inoltre possibile giungere a ulteriori conclusioni. Abbiamo visto che una MT *non dispone delle regole per risolvere un calcolo con le regole di cui dispone*. Questo modo di porre la questione consente di distinguere due diversi livelli di regole presupposte dalle operazioni di una MT: regole relative al tipo e al numero finito di passi da compiere per la risoluzione di un problema (quelle di cui dispone), che potremmo definire euristiche, o *applicative*, e regole relative alla possibilità di risolverlo in quel tipo e numero finito di passi (quelle di cui non dispone), che potremmo perciò definire *condizionali*. Ora, si potrebbe dire che una MT le seconde non le abbia

per il semplice fatto che essa è pur sempre un insieme finito di regole e che, dunque, per definizione essa non possa essere una serie infinita di livelli di regole in cui in ognuno di essi – partendo da quello più in alto – vi siano regole relative alla possibilità di ricorrere a quelle dei livelli più in basso per la risoluzione di un problema. Queste *regole di regole* saranno al limite presenti, oltre un certo livello, nella mente di colui che in base a esse ha programmato la MT che si considera.

Ciò non significa, tuttavia, che la MT creata in questo modo non sia a sua volta creativa. In parte lo abbiamo già mostrato. Abbiamo visto, in particolare, che se una MT non ha le regole a cui attenersi per risolvere un problema con le regole di cui dispone, vuol dire che già solo per il fatto di *iniziare* a risolverlo con *queste* regole essa fa qualcosa che potrebbe fare solamente se avesse (anche) *quelle* regole. Dal momento che invece quelle non le ha, ne consegue che in tal caso essa fa qualcosa di impossibile e dunque di creativo in base alla definizione che abbiamo dato. Con tale argomento, però, la creatività di una MT è dimostrata solamente per *via negativa*, ovvero a partire dalla *mancaza di regole*. Proviamo allora a dimostrarla per *via positiva*. Si è detto che, al limite, solo un soggetto esterno potrebbe avere *in mente*, o *nella tabella* – se ci riferiamo a un'altra MT – le regole in base alle quali ha creato e fornito a una particolare MT le regole di cui dispone. Evitando momentaneamente di stabilire se tale soggetto sia realmente creativo chiediamoci piuttosto se possa esserlo la MT che egli stesso ha programmato. Chiarito questo problema, infatti, si chiarirà anche quello. A prima vista verrebbe da dire che la MT in questione si limiti a eseguire le operazioni per le quali è stata programmata. Questa asserzione coincide perfettamente con la tesi di Lovelace. In base a essa si ricaverebbe facilmente che tale MT non possa essere creativa. Ma si è visto che qualsiasi MT non dispone delle regole relative alla possibilità di risolvere un problema con le regole di cui dispone, e siccome per la tesi di Lovelace – che prima si assumeva come vera – una MT agisce solamente in base alle regole di cui dispone, se poi pur non avendo le regole che le renderanno possibile risolvere un problema con le regole di cui dispone essa alla fine risolverà comunque il problema in questo modo vorrà dire che quelle regole di cui non disponeva avrà dovuto in qualche modo *crearsele*. Chiariamo subito che non ci stiamo spingendo a dire che le avrà effettivamente introdotte nel suo programma e che in base a ciò sarà possibile trovare in esso altre regole – espresse

nel medesimo linguaggio formale – oltre a quelle già inserite dal suo programmatore. Sostenere questo, potremmo aggiungere, significa affermare che una MT è *surrealmente* creativa.¹²

Quello che stiamo sostenendo non è nemmeno che questa *creazione* sia un'operazione formalmente distinta da quella che una MT può eseguire in base alle regole di cui dispone, quanto una sua *necessità interna*. Più precisamente, stiamo sostenendo che forse quando una MT opera in questo modo essa si sta anche *virtualmente* dando le regole per farlo: sta creando, attraverso un *atto istituyente* – cioè senza basarsi su altre regole date – le regole per operare in base a quelle di cui dispone. Lo implica il fatto stesso che essa agisca in base a queste ultime, come se i due atti – l'*applicazione di regole date* e la *creazione di regole nuove* – non fossero altro che i due lati indiscernibili – *attuale* e *virtuale* – di un solo e unico evento. Questo evento implica una creazione che è *reale* per il fatto che non si basa, almeno in parte, sulle regole di cui la MT dispone – motivo per cui sarebbe anche *impossibile*. Che la MT compia tale creazione malgrado sia per essa formalmente impossibile è suggerito da una *necessità paradossalmente interna* al fatto che, come osservava Lovelace, essa compie solo atti che sono per essa formalmente possibili. Questa creazione incide dunque *dall'interno*, in maniera *obliqua*, su questi atti, rendendoli possibili *nel mentre stesso in cui accadono*, non prima – come invece accadrebbe nel caso della creatività trasformativa di Boden. Ma se tale possibilità non viene prima dell'atto che la *realizza* allora essa è sempre già reale in quanto

12. Qualcuno potrebbe obiettare, forse, che le più recenti MT possono creare in un certo qual modo nuove regole – per via del *machine learning* – partendo dai dati che utilizzano. Tuttavia, esse lo fanno (se lo fanno) in quanto sono state programmate per farlo, cioè in base alle regole di cui dispongono (motivo per cui, fra l'altro, non aggiungerebbero realmente delle regole nuove). Ciò a cui qui ci stiamo riferendo è invece una *creazione impossibile* in rapporto alle regole di cui una MT dispone, oppure, in altri termini, una *creazione reale*. La differenza fra questa e una *creazione surreale* starebbe nel fatto che per la seconda una regola la cui creazione non implica il ricorso a regole che la macchina possiede si dev'essere necessariamente scritta all'interno del programma. Dal punto di vista della creazione reale, a fronte del fatto che si sia creata una regola realmente nuova, il fatto poi della sua scrittura nel programma non discende da alcuna necessità logica.

necessità interna all'atto. Essa è, in definitiva, non “il possibile” ma ciò che del reale rimane *impossibile*, in una parola il *virtuale*.¹³

3.2. Ritorno al programmatore

Per via di questa creazione una MT darebbe a se stessa, insomma, le regole delle regole di cui dispone. In tal senso ci si potrebbe anche chiedere se queste metaregole siano identiche, o quanto meno simili, a quelle in base a cui il programmatore pensò le regole interne a MT. Da un punto di vista logico nulla esclude, di fatto, che possa sussistere fra tali due differenti gruppi di metaregole un rapporto di identità o somiglianza. Ma allo stesso modo è anche vero che non vi è nulla che lo certifichi, dal momento che non disponendo delle metaregole del programmatore – essendo queste presenti solamente, al limite, nella mente di esso – la MT non avrebbe potuto di fatto creare le proprie a immagine e somiglianza di quelle, così che tanto l'identità quanto la somiglianza fra le due, eventualmente rilevate a posteriori, si rivelerebbero al tempo stesso delle mere coincidenze. Vi sarebbe piuttosto un altro tipo di analogia, più interessante di questa, rinvenibile non più sul piano del contenuto bensì della *forma* di entrambi gli insiemi di metaregole. Premettiamo che tutte le metaregole del programmatore, come del resto tutte quelle esistenti nella dimensione reale della MT, potranno essere, in base a quanto detto, di due diverse tipologie: metaregole che – come le regole che rendono possibili – sono a loro volta possibili in base ad altre regole; e metaregole che, pur rendendo possibili delle regole e pur agendo simultaneamente alle metaregole del primo tipo, non sono a loro volta possibili – al contrario di queste – partendo da altre regole. Le prime metaregole sono quelle che il soggetto conosce e a cui può ricorrere più o meno consapevolmente per operare, in date situazioni, in base a regole di ordine inferiore. Si pensi ad esempio al caso della *lingua*, cioè un insieme generale di regole reso possibile da altre regole (convenzioni) e a cui un soggetto potrebbe, conoscendolo, attingere, per formulare frasi o segnalare alcune regole *inconsuete* e tuttavia perfettamente possibili dall'interno di esso. Per quanto invece

13. Sulla questione del “possibile” come categoria che non precede logicamente il “reale” si veda (Bergson 2014). Si veda poi (Deleuze 2001) per l'approfondimento della questione bergsoniana e del complicato rapporto fra virtuale e attuale.

concerne le seconde metaregole non si potrà dire che il soggetto le conosca e che potrà ricorrervi allo stesso modo con cui ricorre alle prime. Assumiamo allora, anche al fine di comprenderle meglio, che le regole in base a cui il programmatore sopra menzionato pensò le regole di cui la MT disponeva fossero dello stesso tipo di quelle che la MT creò *virtualmente* operando con le sue regole, ovvero metaregole del secondo tipo. In questo caso le metaregole del programmatore non potranno essere state delle regole a cui egli avrà potuto fare ricorso ma che avrà dovuto *creare* per eseguire le operazioni possibili in base alle regole di cui disponeva, con la sola differenza di aver trasferito queste ultime dalla sua dimensione a quella della MT per farle svolgere le suddette operazioni al posto suo. Ma a questo punto, a ben vedere, siamo anche in grado di svincolarci dalla dimensione della MT, per pensare la stessa questione, ovvero le condizioni della creatività reale, all'interno della dimensione umana. Questo ci dà il vantaggio di poter accedere a un punto di vista maggiormente descrittivo su ciò la cui esistenza abbiamo fin ora inferito astrattamente su basi logiche. Chiediamoci allora quali tipi di atti umani potrebbero essere ritenuti conformi alla definizione di creatività reale. Si prenda a tal fine, di nuovo, il caso della lingua. Grazie a essa è possibile eseguire una serie di operazioni come leggere, scrivere o parlare, trattando con un numero finito di *segni*. Per leggere, ad esempio, devo guardare un testo e decodificarne i segni. Se riesco a fare questa decodifica vuol dire che essi, oltre che segni, sono *simboli* di una lingua che conosco. Ma affinché avvenga questa decodifica, affinché io possa applicare le regole della lingua ai segni che vedo, devo anche simultaneamente poter trattare quei segni come se fossero casi particolari dei simboli che conosco, devo creare cioè una regola, nel mentre stesso della decodifica, che non può essere a sua volta presente nell'insieme delle regole della lingua che conosco e per la quale stabilisco questa stessa cosa – ovvero che una serie di segni che vedo corrisponde a una serie di simboli che conosco – al fine di rendermi possibile la lettura del testo. Questo oscuro lavoro avverrebbe in quella dimensione interna e assieme obliqua al *simbolico* che è l'*immaginario*. Esso implica ciò che Castoriadis definì nei termini di *un quid pro quo* interno al «*linguaggio* inteso come codice» (2022, 371, 381-399): una serie di operazioni con cui – nel mentre stesso della decodifica e senza ricorrere a ulteriori regole interne alla lingua che la renderebbero possibile – si prende *una cosa per un'altra* – un segno

qualsiasi, in questo caso, per un simbolo – in maniera tale che le regole della suddetta lingua diventino disponibili per operazioni come leggere, scrivere o parlare.

Tali micro operazioni riproporrebbero inoltre, in ogni singolo passaggio di livello che è possibile fare fra quelli che definiscono le regole di un sistema simbolico, l'*operazione fondamentale* che l'ha reso possibile all'inizio, ovvero l'insieme delle decisioni con cui si è potuto stabilire, ad esempio, che la sequenza di segni "c-a-n-e" rinverrà sempre all'immagine di un certo animale a quattro zampe e non a quella di un albero, e che prese di per sé non avrebbero alcun senso – proprio in quanto cominceranno ad averne uno, assieme alla serie infinita di mondi che renderanno possibili, solo una volta che verranno prese –, o non più di quanto potrebbero averne le associazioni con cui si è abituato a convivere uno schizofrenico.¹⁴ In entrambi i casi, al fine di fissare un accordo di qualche tipo tra il proprio mondo e i vincoli esterni, tra l'esigenza di esprimersi, ad esempio, e l'impossibilità di riuscirci senza un linguaggio, si verrà meno in qualche modo alla ragione, si dirà senza seguire alcuna logica che *qualcosa significa qualcos'altro*.

Conclusione. Si può dire che i computer immaginano?

Qualcuno potrebbe anche pensare, forse a ragione, che questa forma di creatività non sia nulla di *speciale* in quanto fiorisce pur sempre nell'alveo del compromesso con una serie di regole date. A patto di intendersi sul fatto che ciò che da quell'alveo così fiorisce, per quanto compromesso, non poteva essere in alcun modo deducibile da ciò che nell'alveo era già dato, e che dunque è, una volta fiorito, del tutto *nuovo e realmente creativo*, non faremo nulla per convincervi del contrario. D'altra parte, non era di nostro interesse trattare qui di casi straordinari di creatività – anche perché questi rientrerebbero probabilmente nell'insieme della creatività apparente (non perché non sarebbero creativi o straordinari, ma letteralmente perché, in tal caso, sarebbero così per qualcuno e non in quanto tali). Quello che si è cercato di fare, analizzando la

14. Per un approfondimento dei casi di schizofrenia in quanto rivelativi di una forma di creatività che accomuna l'uomo e la macchina si veda *infra* Cardone L. (2023), *Immagini schizofreniche. Riflessioni fenomenologiche sulla creatività computazionale a partire da Marc Richir*.

creatività al cuore del compromesso con l'insieme delle regole date, è stato piuttosto mostrare, da una parte, quelle che si potrebbero definire le *condizioni minime* della creatività reale, e dall'altra come vi sia un *marginale minimo* di creatività reale in qualunque atto reso possibile da regole già date. Questo atto è in tal senso sempre già spaccato in due metà asimmetriche dallo slancio circolare con cui una vita rifà se stessa nel medesimo istante in cui è chiamata a ri-farsi carico di regole stabilite. Non ci è nemmeno sfiorata l'idea, inoltre, di trattare da un punto di vista *descrittivo* la creatività reale nelle macchine di Turing – ovvero nei computer per come ancora oggi li conosciamo. Ci piace sicuramente pensare che vi sia in essi qualcosa come il corrispettivo di ciò che nell'uomo chiamiamo immaginario. Ma questo, appunto, non è altro che un pensiero in libertà. Siamo persuasi, invece, che per quanto fin qui sostenuto vi debba essere in essi una sorta di dimensione che quanto meno *affianca* – se non *fonda* – quella simbolica all'interno di ognuna delle operazioni che eseguono. Ritornando sulla sponda dell'umano, verrebbe da pensare a quei frangenti in cui, quando si calcola a mente, ognuno di noi crea tutto un mondo di immagini e linee che fanno da supporto a una serie di operazioni formali. Ecco: lì, in quel preciso istante, facciamo qualcosa di incomunicabile sul piano simbolico e che, malgrado ciò, accompagna la manipolazione di simboli funzionale alla risoluzione di un calcolo. Qualcuno potrebbe aver inventato dei percorsi tortuosi, complicatissimi, che gli farebbero fare dei giri infiniti prima di giungere a una soluzione. Percorsi inutili, dunque, ma indubbiamente unici nel loro genere e magari interessanti. Altri potrebbero riprodurre nella mente gli stessi schemi o disegni imparati la prima volta a scuola per fare le operazioni sul quaderno, e anche in questi casi ci vorrà comunque creatività – ma appunto una creatività di cui ognuno di noi è dotato in partenza, e che per questo motivo potrebbe anche non soddisfare qualcuno – per riprodurre nella mente cose di per sé adeguate a essere scritte su un foglio di carta. Ma la domanda è: cosa fa un computer in questi casi? Cosa fa, cioè, *mentre* calcola?

Si parla tanto della componente *software* e della sempre maggiore complessità dei nuovi programmi in circolazione. Si parla tanto, anche, della componente *hardware*, e con essa delle analogie con il funzionamento del cervello umano. Noi ci chiediamo: è possibile parlare – se è sostenibile che essi siano capaci di creare nel senso fin qui descritto –

di un immaginario e di un'immaginazione propri dei computer? Per rispondere a una domanda come questa – ma anche a quella, in fondo, relativa alla creatività computazionale – bisognerebbe forse essere in grado, fra le altre cose, di descrivere il corrispettivo di quei mondi di immagini e linee degli umani a livello dei computer. Ma su questo sarà necessario lasciare il passo a un'indagine maggiormente descrittiva dei processi computazionali. Un'indagine per la quale potranno non essere sufficienti, effettivamente, le definizioni generali e oggettive, ma che di esse potrebbe aver bisogno, appunto, per riconoscere le condizioni minime di qualsiasi operazione creativa. Potrebbe tornare utile, da questo punto di vista, riprendere in considerazione la questione della creatività apparente, cioè quei casi di computer che fanno cose sorprendenti dal punto di vista di un umano che li osserva, in quanto potrebbero essere il portale di accesso a un modo di creare e di immaginare ancora sconosciuto all'umano – per quanto forse idealmente spiegabile, dal nostro punto di vista, con l'idea di creatività reale.

Riferimenti bibliografici

Arieti, S. 1986, *Creatività. La sintesi magica*, Il pensiero Scientifico Editore, Roma.

Beaudot A. 1977, *La creatività*, Loescher, Torino.

Bergson H. 1984, *Il possibile e il reale*, trad. di F. Sossi, in "Aut-Aut", 204, pp. 3-13.

Boden M. A. 2004, *The Creative Mind. Mith and mechanisms*, Routledge, Londra-New York.

Boden M. A. 1987, *Artificial intelligence and natural man*, The MIT Press, London.

Cantone D. 2022, *La filosofia degli automi. Origini dell'intelligenza artificiale*, Mimesis, Milano-Udine.

Castoriadis C. 2022, *L'istituzione immaginaria della società*, trad. di E. Profumi, Mimesis Edizioni, Milano-Udine.

Copeland J. 2004, *The Essential Turing. Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*, Oxford University Press, Oxford.

Dartnall T. 1994, *Artificial Intelligence and Creativity. An Interdisciplinary Approach*, Springer Science+Business Media Dordrecht, Brisbane.

Davis M. 1958, *Computability and Unsolvability*, MacGraw-Hill Book Company, New York.

Deleuze G. 2001, *L'immagine-tempo. Cinema 2*, trad. di L. Rampello, Ubulibri, Milano.

Frixione M. & Palladino D. 2017, *La computabilità: algoritmi, logica, calcolatori*, Carocci Editore, Roma.

Gangemi A., Miceli S., Sprini G. 2003, *L'intelligenza. Teorie e modelli*, Laterza, Roma-Bari.

Gödel K. 1931, *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I*, «Monatshefte für Mathematik und Physik», 38, pp. 173-198.

Guilford J. P. 1950, *Creativity*, *American Psychologist*, 5, (9), pp. 444-454.

Hodges A. 2012, *Alan Turing: The Enigma*, Princeton University Press).

Lucas S. 2021, *The origins of the halting problem*, "Journal of Logical and Algebraic Methods in Programming", 121,100687.

Menabrea L. F., Lovelace A. A. 1842, *The Analytical Engine. Invented by Charles Babbage, With notes upon the Memoir by the Translator Ada Lovelace, Countess of Lovelace*, Bibliotheque Universelle de Geneve, 82, October.

McCarthy J., Minsky M. L., Rochester N., Shannon C. E. 2006, *Proposta di un Progetto di ricerca estivo sull'intelligenza artificiale presso il*

Dartmouth College, "Sistemi intelligenti", XVIII, 3, pp .413-428.

Trombetta C. 1990, *La creatività. Un'utopia contemporanea*, Bompiani, Milano.

Turing A. M. 1950, *Computing machinery and Intelligence*, "Mind", 59, pp. 433-460.

Turing A. M. 1936, "On computables numbers, with an application to the Entscheidungsproblem", Princeton University Press, Princeton.

Veale T., Almicar Cardoso F. (2019), *Computational Creativity. The Philosophy and Engineering of Autonomously Creative Systems*, Springer, Cham Switzerland.

Wallas G. 2014, *The art of thoughts*, Solis Press, Kent England.

Michele Pavan
Università degli Studi di Verona
michele.pavan@univr.it