“Ignoro, dunque sono”

Germano D’Abramo

1. *Introduzione*

La possibilità di riprodurre artificialmente la mente umana ha affascinato i maggiori pensatori di tutte le epoche, soprattutto filosofi e matematici, ma recentemente anche informatici, biologi e neurologi. Tutti hanno cercato di fornire argomenti a favore o contro questa ipotesi con gli strumenti a loro disposizione. È inutile dire che la letteratura esistente a tal proposito è sterminata e che su tale possibilità un consenso tra gli studiosi è ancora là da venire. In questo articolo vogliamo discutere un aspetto specifico di questa problematica, che ovviamente non esaurisce tutte le possibilità, e che potrebbe andare sotto il nome di *riproducibilità algoritmica* della mente umana.

Esiste infatti almeno un’altro tipo di riproducibilità della mente oggetto di discussione e di studio, ed è la *riproducibilità fisica*: in sostanza, se si riuscisse a riprodurre fisicamente e funzionalmente (cioè nelle sue funzioni) il cervello umano, magari con materiali diversi da quelli che lo costituiscono in natura, si potrebbe essere certi di aver riprodotto la mente umana? Posso assicurare che questo interrogativo dà origine ad innumerevoli altre questioni, anche di natura filosofica, e genera paradossi difficilmente maneggiabili (si veda, ad esempio, D’Abramo 2007).

Qui invece vogliamo occuparci della possibilità di esistenza di un algoritmo, cioè di una procedura o sequenza di operazioni prefissate, che possa simulare le attività (le funzioni) del cervello e in ultima analisi simulare la mente umana (o coscienza).

Per intenderci, algoritmo è ad esempio ogni programma per computer o regola per la risoluzione di una operazione aritmetica, ma è anche, in senso lato, l’insieme delle istruzioni che dobbiamo fornire per far funzionare correttamente una qualsiasi macchina programmabile. Ovviamente una sequenza di istruzioni, per quanto lunga e complessa, non basta da sola a riprodurre la mente umana; se ciò fosse vero, sarebbe come affermare che i fogli di carta su cui questa sequenza è scritta possono “pensare”. Ad ogni algoritmo deve

Chaos e Kosmos – www.chaosekosmos.it
Rivista online
ISSN 1827-0468
Autorizzazione del Tribunale di Roma nr. 320/2006 del 3 Agosto 2006
Direttore responsabile e proprietario Riccardo Chiaradonna

quindi essere associata una macchina in grado di "eseguire" queste istruzioni, conferire loro l'aspetto dinamico che caratterizza anche i processi cerebrali e mentali dell'uomo. La più semplice e generale di queste macchine è la famosa "macchina di Turing"[1], che è, per intenderci, il modello matematico alla base di ogni calcolatore moderno.

In questo contesto non è indispensabile sapere in dettaglio cosa sia una macchina di Turing e come funzioni. Basta immaginarla come un moderno calcolatore in grado di mandare in esecuzione ogni algoritmo che scriviamo nella sua memoria in un prefissato linguaggio.

A questo punto immaginiamo che in alcuni dei lettori inizi a prendere corpo una sorta di insofferenza, legata al fatto che per loro è già evidente che la mente umana non può essere riprodotta da alcun algoritmo. Sebbene anche l'autore nutra intimamente questa convinzione, in questi casi le intime convinzioni possono essere forvianti. A dimostrazione del fatto che la problematica di questo articolo non sia affatto banale ci sono anni e anni di ricerca scientifica e filosofica proprio sull'aspetto della riproducibilità algoritmica della mente[2]. L'argomento descritto in questo articolo è stato sviluppato originariamente in D'Abramo 2005, tuttavia qui abbiamo voluto indugiare di più su un aspetto interpretativo che ha un sapore più filosofico che matematico e che è ben sintetizzato nel titolo dall'esplicito riferimento alla famosa frase di Cartesio.

Ad essere rigorosi, tuttavia, il titolo per esteso avrebbe dovuto essere: "So di ignorare, dunque sono". La mancanza di informazione, l'ignorare appunto, ci spinge necessariamente a ragionare in termini probabilistici (e qui entra in gioco il "So di ignorare...") e a fare delle assunzioni di base, come, ad esempio, l'ipotesi dell'equiprobabilità di un insieme di eventi quando non abbiamo a priori alcun motivo di pensare diversamente, che unite alla nozione di complessità algoritmica ci permettono di presentare un solido e convincente argomento[3] contro la riproducibilità algoritmica della mente umana.

---

[1] La macchina di Turing prende il nome dal matematico inglese Alan M. Turing che l'ha proposta nella prima metà del secolo scorso.

[2] Uno dei più famosi e controversi argomenti contro la riproducibilità algoritmica della mente e della coscienza umane è quello della Stanza Cinese del filosofo americano John R. Searle (si veda, ad esempio, *Minds, Brains, and Programs*, da *The Behavioral and Brain Sciences*, Vol. 3, Cambridge 1980).

[3] Anche se non potrà mai appartenere alla famiglia delle dimostrazioni rigorose.

2. *La complessità algoritmica*

La nozione di complessità algoritmica (o di Kolmogorov) è una nozione non molto difficile da comprendere oggigiorno, data l'enorme diffusione dei calcolatori elettronici. Dato un numero binario s, cioè una stringa di zero e di uno, la complessità algoritmica di s è definita come l'estensione in bit[4] del più piccolo programma per computer in grado di generarla.

In simboli, fissato per convenzione un arbitrario linguaggio di programmazione e un arbitrario computer U (macchina di Turing) capace di eseguire un programma scritto nel suddetto linguaggio[5], la complessità algoritmica $H(s)$ della stringa s è definita come segue:

$$H(s) \equiv \min_{U(p)=s} |p|, \tag{1}$$

dove p è la stringa di zero e uno che codifica il programma capace di generare s tramite U. In sostanza, la complessità $H(s)$ è la lunghezza del più piccolo programma p in grado di generare la stringa s (per una definizione più rigorosa, si veda Chaitin 1975, 1987).

La definizione di complessità algoritmica è ricca di risvolti matematici importanti, quali, ad esempio, una rilettura originale del problema dell'arresto di Turing e una nuova ed interessante caratterizzazione di stringa casuale o random (si veda ancora Chaitin 1975, 1987).

Da quanto è stato detto sia p, il programma da fornire in ingresso al computer U, che s, il numero binario prodotto da p

---

[4] Un bit (contrazione di **bi**nary digi**t**) è una cifra binaria, ovvero uno dei due simboli del sistema numerico binario, classicamente chiamati zero (0) e uno (1). Dal punto di vista della teoria dell'informazione il bit è anche la più piccola unità di informazione che può essere trasmessa (ad esempio, vero o falso, si o no, etc.). In informatica esso è anche l'unità di misura base dell'estensione dei programmi per computer: in linea di principio, infatti, nel linguaggio macchina ogni programma è una stringa di zero e di uno.

[5] La tipologia specifica del linguaggio di programmazione e del computer in grado di eseguire programmi scritti in tale linguaggio è ininfluente ai fini della definizione di complessità algoritmica. Una volta stabiliti per convenzione un linguaggio e un computer in grado di eseguirlo, la complessità algoritmica di una stringa ha un valore intrinseco e può essere stabilita la complessità relativa fra due o più stringhe.

attraverso U, sono delle stringe binarie. Inoltre, ad ogni programma in ingresso p corrisponde, tramite U, una sola stringa di uscita s. Può anche succedere che per un determinato programma p, $U(p)$ non produca nessuna stringa di uscita s (anche se in generale non siamo in grado di dimostrare rigorosamente che non la possa produrre; questo è il famoso *problema dell'arresto* di Turing). In questo caso la complessità di s è definita *infinita* per convenzione. A questo punto è utile far notare che il numero di programmi esistenti con estensione minore o uguale a N bit è pari a $2^{N+1}-2$. Il numero di programmi di un bit, infatti, è 2 (cioè 0 e 1), il numero di quelli di due bit è 2^2 (cioè 00, 01, 10 e 11), il numero di quelli di tre bit è 2^3, e così via. Quindi, il numero di programmi di estensione minore o uguale a N bit è

$$\sum_{i=1}^{N} 2^i = 2 + 2^2 + 2^3 + ... + 2^{N+1} = 2^{N+1} - 2 \quad ,$$

Ad esempio, tutti i programmi di un numero di bit minore o uguale a 2 sono: 0, 1, 00, 01, 10, 11. Cioè, sono appunto $2^3 - 2 = 6$.

Quindi, il massimo numero di stringhe distinte s che possono essere prodotte come uscita da programmi di estensione minore o uguale a N bit è proprio[6] $2^{N+1}-2$. Una conseguenza ovvia, ma interessante per il seguito di questo articolo, della proprietà appena espressa è che se consideriamo tutte le stringhe s di $N+k$ bit (in totale sono $2^{N+k+1}-2$) sappiamo per certo che alcune di queste non possono essere prodotte da nessun programma p di estensione minore o uguale a N bit, poiché $N < N+k$ e quindi $2^{N+1}-2 < 2^{n+k+1}-2$, cioè non vi è un numero sufficiente di programmi p di estensione minore o uguale a N bit che le possa produrre tutte.

Inoltre, è chiaro dalla definizione (1) che la complessità algoritmica di una qualunque delle $2^{N+1}-2$ stringhe s non può essere superiore a N bit, essendo esse state generate da programmi che hanno una estensione massima di N bit.

Nella prossima sezione vedremo come la definizione di complessità algoritmica e le proprietà ad essa associate ci forniscano interessanti spunti di riflessione sulla riproducibilità algoritmica della mente umana.

---

[6] A beneficio del lettore ricordiamo che ad un programma p corrisponde al massimo una singola stringa di uscita s.

3. *Complessità algoritmica e mente umana*

Facendo uso degli strumenti matematici introdotti nel paragrafo precedente, in questa sezione intendiamo fornire un convincente supporto alla seguente tesi:

La probabilità che il cervello, più precisamente la sua funzione di dare origine alla mente e alla coscienza, sia simulabile e riproducibile attraverso un algoritmo finito (ad esempio, un algoritmo di N bit, anche con N molto grande) è arbitrariamente vicina a 0.

Nel linguaggio matematico dimostrare un'affermazione come la precedente significa a tutti gli effetti fornire una dimostrazione matematica dell'impossibilità di riprodurre la mente umana attraverso un algoritmo. Tuttavia, quanto noi proporremo qui non ha la presunzione di essere una dimostrazione rigorosa, cerca di essere piuttosto un forte argomento a favore della suddetta tesi.

Per il gusto dell'esperimento pensato, supponiamo che i processi cerebrali che danno origine al pensiero e alla mente (e quindi all'esperienza cosciente) possano essere simulati e riprodotti da un algoritmo di N bit (dove N può essere un numero arbitrariamente grande). Allora saremmo in grado di riprodurre algoritmicamente le funzioni del cervello, e quindi la mente, come un programma che viene eseguito da un computer, quindi da un supporto differente da quello biologico/organico del corpo umano e in linea di principio capace, a differenza di questo ultimo, di funzionare per un tempo illimitato. Inoltre, immaginiamo che come parte del programma che riproduce la mente ci sia un semplice (dal punto di vista della complessità algoritmica) e veloce contatore capace di enumerare (ma non memorizzare, altrimenti lo spazio di memoria fisica potrebbe non essere sufficiente) tutti i numeri binari, ad esempio di c bit (ve ne sono 2^c in totale), in ordine lessicografico, cioè nel classico ordinamento numerico crescente, 0, 1, 00, 01, 10, 11, 000 e così via.

Ora, la mente simulata attraverso l'algoritmo, per ipotesi assimilabile in tutto e per tutto ad una mente umana, sarebbe capace di immaginare un numero decimale più grande dell'estensione (in bit) del suo proprio algoritmo generatore, diciamo $N+k$, ed avviare il contatore interno in maniera tale che enumeri in ordine crescente tutte le possibili stringhe (numeri) binarie di estensione pari a $N + k$ bit (ve ne sono 2^{N+k}), memorizzando solo l'ultima in ordine di tempo.

A questo punto, la mente algoritmica potrebbe scegliere (come ogni essere umano sarebbe in grado di fare) di interrompere l'enumerazione ad un istante arbitrario e stampare l'ultimo numero raggiunto in ordine di tempo.

Il contatore interno può essere provvisto di un indicatore di completezza capace di dare la percentuale dell'intero conteggio raggiunta fino a quel momento. Questo accorgimento potrebbe essere di qualche utilità allo scopo di scegliere indicativamente quando fermare il conteggio; non troppo presto, ad esempio, dal momento che le prime stringhe binarie enumerate non sono certamente molto complesse dal punto di vista algoritmico (questo aspetto sarà più chiaro fra breve).

A questo punto ricordiamo che il numero stampato è una stringa binaria di $N + k$ bit (ordinalmente più piccola dell'equivalente binario del numero 2^{N+k}, che rappresenta l'estremo superiore del conteggio) e inoltre essa risulta in qualche modo scelta casualmente o, per meglio dire, risulta scelta alla cieca dalla mente algoritmica. In sostanza, la mente algoritmica ignora tutte le caratteristiche del numero da essa scelto, così come accadrebbe ad ogni mente umana se dovesse operare la stessa scelta in condizioni analoghe.

Quindi, con una probabilità P circa uguale a

$$P = 1 - \frac{2^{N+1} - 2}{2^{N+k}} \approx 1 - 2^{1-k} ,$$

(2)

dove $2^{N+1} - 2$ è il numero totale di stringhe binarie distinte che possono essere ottenute da un numero di programmi di estensione minore o uguale a N bit[7], l'algoritmo che simula la mente sarebbe capace di produrre in uscita una stringa di complessità più grande di N bit. E per valori di k molto maggiori di 1 questa probabilità diventa arbitrariamente vicina ad 1.

Il rapporto $(2^{N+1} - 2)/(2^{N+k})$ nell'equazione (2) rappresenta proprio il numero di programmi distinti esistenti con estensione minore o uguale a N bit, e quindi il numero massimo di stringhe binarie che possono

[7] Cioè, l'estensione dell'algoritmo che riproduce la mente.

essere generate da tali programmi, sul numero totale di stringhe di estensione pari a $N+k$, fra le quali la mente algoritmica può scegliere alla cieca un numero binario da stampare (e proprio il fatto che la scelta sia alla cieca permette di scrivere la probabilità di scelta come il precedente rapporto, secondo la teoria classica della probabilità).

Questi ultimi passaggi sono sicuramente la parte più difficile da seguire. Ricapitoliamo allora. Nell'ipotesi che il cervello (e quindi, la mente) sia riproducibile da un algoritmo di estensione pari a N bit, la probabilità che tale algoritmo sia capace di generare una stringa binaria più complessa di N bit è arbitrariamente vicina ad 1. E questo conduce direttamente ad un paradosso logico insolubile, data la definizione di complessità algoritmica. Dunque, la probabilità che l'ipotesi di partenza (la riproducibilità algoritmica della mente) sia violata è arbitrariamente vicina ad 1.

In sostanza, il semplice fatto che la mente algoritmica ignora la complessità della stringa che va a scegliere e a stampare[8], conduce inevitabilmente ad un paradosso logico, poiché si può rendere la probabilità che tale stringa sia più complessa dell'algoritmo che la produce (cioè, che la sceglie e la stampa appunto), arbitrariamente vicina ad 1.

Ovviamente si può avanzare una serie di obiezioni all'argomento sopra esposto, tutte alquanto ragionevoli. Si può affermare che per grandi valori di N l'enumerazione prima descritta (che coinvolge 2^{N+k} numeri binari di $N+k$ bit) non sia fisicamente possibile, poiché richiederebbe una quantità di tempo incredibilmente grande. A carattere esemplificativo, si potrebbe tentare una stima e scoprire che, volendo rimanere conservativi, sarebbe necessario un tempo di gran lunga superiore all'età stimata dell'Universo.

Alcuni, inoltre, potrebbero affermare che il precedente argomento non elimina a priori la possibilità che la nostra mente sia "algoritmica": potremmo essere simili a "macchine" e comportarci prevedibilmente come "macchine" (magari scegliendo inconsapevolmente, nel caso della procedura sopra descritta, una stringa di complessità algoritmica consona, per cui la relazione (2) perderebbe ogni significato), ma essere semplicemente e in maniera errata convinti del contrario.

Tuttavia, anche volendo prendere in seria considerazione queste critiche, il nostro argomento dovrebbe mantenere un suo interesse:

[8] Così come succederebbe alla mente umana in condizioni analoghe.

malgrado quanto esposto possa essere fisicamente non fattibile, la correttezza logica e matematica del nostro argomento rimane inalterata. D'altro canto, molte dimostrazioni matematiche considerate ineccepibili dalla comunità dei matematici sono fisicamente non verificabili poiché , ad esempio, coinvolgono il concetto di infinito (attuale o potenziale che sia).

4. *Considerazioni conclusive*

In questa sezione conclusiva vogliamo suggerire una possibile interpretazione dei risultati sopra mostrati. Per farlo partiamo da un'altro esperimento pensato. Prendiamo in considerazione un dispositivo meccanico che legge in ingresso un numero decimale N e lancia una moneta, non truccata, N volte.

Ogniqualvolta il risultato del lancio è testa questo dispositivo stampa un 1 su un lungo nastro di carta, altrimenti stampa uno 0. Se si analizza questo strumento dal punto di vista della complessità algoritmica si è tentati di affermare che l'estensione algoritmica di questo dispositivo sia proporzionale a $\log_2 N$, cioè al numero di bit[9] dell'espansione binaria del numero decimale N. Se, in qualche modo, immaginiamo che il dispositivo abbia una estensione algoritmica intrinseca costante, pari a k bit, l'estensione totale di questo algoritmo, dopo che abbiamo fornito in ingresso e fatto memorizzare il numero decimale N, sarà pari all'estensione intrinseca k più l'estensione del numero decimale stesso, $\log_2 N$. In totale $\log_2 N + k$, che è proprio una quantità proporzionale a $\log_2 N$.

Ora, poiché anche questo dispositivo sceglie casualmente le cifre della sua stringa di uscita (quale procedura migliore per farlo se non il lancio di una moneta?), se noi diamo in ingresso un numero N opportunamente grande, tale che $N >> \log_2 N + k$, con un calcolo simile a quello svolto nell'equazione (2) avremo che il dispositivo è in grado di generare una stringa di complessità superiore alla propria estensione con una probabilità arbitrariamente vicina a 1. In questo

---

[9] Il numero di cifre 0/1 dello sviluppo binario di un numero decimale N è pari a $\lfloor \log_2 N \rfloor + 1$, dove $\lfloor x \rfloor$ è l'approssimazione all'intero più piccolo di x. Ad esempio, per $N = 27$, $\log_2 27 \approx 4.75$, quindi $\lfloor \log_2 27 \rfloor + 1 = 5$ e la rappresentazione di 27 in base binaria è $(27)_2$ = 11011, che ha proprio 5 cifre binarie.

caso, tuttavia, l'estensione algoritmica del dispositivo è comparabile alla complessità dell'intero processo fisico del lancio di una moneta, di tutte le leggi fisiche coinvolte e di tutte le relative condizioni iniziali che fanno si che da un singolo lancio risulti testa piuttosto che croce, o viceversa. In sostanza, quindi, il dispositivo appena descritto non è un algoritmo confinato in un computer; esso, piuttosto, è un dispositivo fisico che opera sotto l'influenza del mondo circostante, delle sue leggi fisiche e delle sue interazioni.

Forse lo stesso argomento che suggerisce una soluzione per il paradosso della complessità del nostro dispositivo meccanico vale anche per la mente umana. La peculiarità del cervello umano nel dare origine alla mente e all'esperienza cosciente potrebbe avere origine proprio nella sua complessa e continua interazione fisica con il mondo circostante. Come voler dire che la nostra mente non può avere origine al di fuori dell'Universo fisico a cui appartiene[10], nel senso delle leggi fisiche che lo governano, ma anche nel senso delle interazioni fisiche possibili e ineliminabili del nostro cervello con tutto ciò che l'Universo contiene (a questo proposito si veda, ad esempio, D'Abramo 2007).

BIBLIOGRAFIA

Chaitin 1975: G.J. Chaitin, *Randomness and mathematical proof*, Scientific American 232 (May 1975), pp. 47-52

Chaitin 1987: G.J. Chaitin, *Algorithmic Information Theory*, Cambridge 1987

D'Abramo 2005: G. D'Abramo, *Some non-conventional ideas about algorithmic complexity*, Chaos, Solitons & Fractals 25/1 (2005), pp. 29-32. Preprint arXiv:math/0211222v8 (18 Mar. 2005)

D'Abramo 2007: G. D'Abramo, *A thought experiment on consciousness*, NeuroQuantology Journal 5(4) (2007), pp. 392-395. Preprint arXiv:physics/0310154v5 (5 Sett. 2005)

[10] Ad esempio, come procedura formale/simbolica eseguita da un computer.