

Appunti di intelligenza artificiale

Guida ad una antica questione e ad una nuova disciplina.

Alan Turing e mio marito parlavano della macchina e di ciò che avrebbe fatto in futuro. Alan stava dicendo meditabondo: «Quando arriverà a quel punto, credo non sapremo più come lo fa».
(da una lettera della moglie di M.H.A. Newman a Sara Turing)

In buona misura, la delimitazione tra intelligenza artificiale e computer science in generale è più una questione di intenti e interpretazioni che non di precise distinzioni di campo. I primi passi dell'elaboratore elettronico sono stati fortemente condizionati da problemi teorici di logica, di matematica razionale e di calcolo proposizionale, e ciascun informatico che si rispetti si è confrontato, prima o poi, col tema della «macchina che pensa». Tuttavia, anche a prescindere da compartimentazioni recenti, quali la robotica, il lavoro sul linguaggio naturale o l'interpretazione delle immagini, vi sono alcune attività che ormai è tradizione includere nel corpo dell'ia classica e che possono essere raggruppate intorno a tre principali strategie di ricerca: le tecniche per la soluzione dei problemi, lo sviluppo delle macchine astratte e l'utilizzo delle regole di produzione.

La soluzione di problemi

Le condizioni fondamentali per risolvere un problema sono:

- 1) che la soluzione esista;
- 2) che sappiamo riconoscerla quando ci imbattiamo in essa;
- 3) che abbiamo un metodo per esplorare sempre nuove possibilità, per evitare di girare in tondo a provare e scartare quelle che abbiamo già provato e scartato.

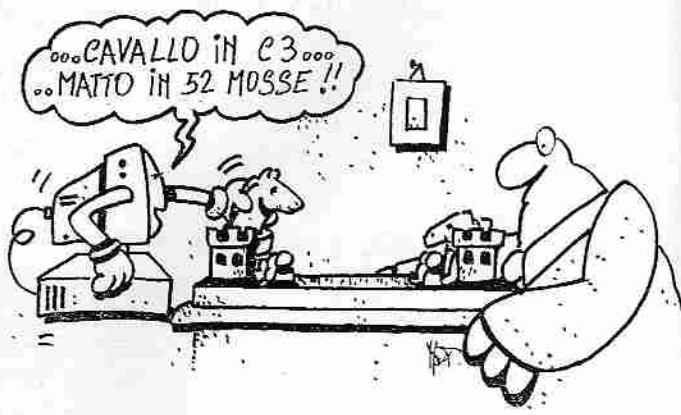
Con una metafora topologica, si dice che un problema individua uno spazio di ricerca comprendente (si spera) almeno una soluzione soddisfacente; affrontare un problema significa esplorare con una qualche sistematicità il terreno così definito, verificando se ciò che incontriamo nel cammino soddisfa o meno i criteri che ci siamo dati per

L'intelligenza artificiale classica.

riconoscere una «buona» soluzione. In pratica, poiché non si tratta di uno spazio oggettivo ma di uno spazio per così dire virtuale, che noi stessi creiamo nel momento in cui precisiamo e formalizziamo il problema, si preferisce individuare in primo luogo una funzione che costruisca possibili risposte, e in secondo luogo una funzione che giudichi della loro efficacia: un modulo di generazione ed un modulo di valutazione. Il terreno più propizio è quello di giochi astratti quali il filetto, gli scacchi o la dimostrazione di teoremi di logica proposizionale, perché rispondono abbastanza bene alle tre condizioni sopra enunciate. Per esempio, in una partita a scacchi lo scopo è la cattura del re avversario, facilmente riconoscibile quando finalmente avviene; in ogni momento del gioco, è possibile elencare tutte le mosse lecite e sperimentarle sistematicamente, sviluppando le differenti linee di gioco fino a scegliere quelle che conducono all'esito prefissato. Lo spazio di ricerca è dato da tutte le possibili partite che dallo schieramento iniziale portano alla cattura di uno dei re o ad una patta (negli scacchi o nella dama non esistono partite senza fine, perché al terzo riproporsi di una stessa disposizione la patta è automatica). La soluzione del problema consiste allora nello scegliere una partita che conduca alla vittoria.

Tastiera e scacchiera

Una prima difficoltà, tipica dei giochi competitivi (a differenza di un rompicapo o di un solitario), è che anche l'avversario vuole vincere e quindi sovrappone le sue scelte a quelle che spettano a noi. Un algoritmo soddisfacente deve perciò tener conto delle possibili contromosse dell'altro giocatore, «immedesimandosi» nel suo punto di vista: in pratica, la strategia è applicata due volte, alle proprie e alle altrui mosse, supponendo cioè che anche l'avversario giochi al meglio. Questo permette di scartare non solo le mosse disastrose per i propri pezzi, ma anche quegli sviluppi che conseguirebbero da risposte fallimentari dei pezzi nemici. Se immaginiamo l'albero completo dei possibili sviluppi di una partita a scacchi (un albero incredibilmente denso e frondoso, perché ad ogni mossa del bianco o del nero si creano in media una ventina di biforcazioni, e altrettante per ognuna di esse, fino ad oltre 100 semimosse per partita!), possiamo vedere il nostro obiettivo come l'individuazione di un cammino ottimale verso la cattura del re. Tale strategia è chiamata «ricerca minimax», perché è data dall'alternanza di una fase di «massimizzazione» del vantaggio, ovvero dalla scelta della mossa che ha una più alta valutazione rispetto al nostro obiettivo, e di una fase di «minimizzazione» che simula la contromos-



sa dell'avversario, corrispondente all'alternativa a noi più sfavorevole. Attenendosi alle regole formali di base e con la sola capacità di riconoscere la soluzione al momento della cattura del re, l'algoritmo di ricerca non è in grado di fornire alcuna valutazione di una mossa fino a quando non ha esplorato tutte le sue più remote conseguenze. Il paradosso consiste nel fatto che per trovare la mossa più vantaggiosa dobbiamo in effetti provarle tutte (almeno fino a quando non scopriamo un percorso sicuramente vincente e che quindi ci soddisfa). In altre parole, prima ancora di effettuare la prima mossa dovremo giocare «mentalmente» tutte le partite possibili e poi scegliere di conseguenza. Sfortunatamente, nel caso degli scacchi il numero di partite possibili è talmente al di là di ogni immaginazione (nella nostra ipotesi, una media di 20 alternative per mossa elevato a 100 semimosse per partita) da rendere del tutto inapplicabile un metodo esaustivo di questo tipo, se non forse per lo studio dei finali o la soluzione di problemi tipo: «il bianco muove e vince in due mosse».

L'arte della potatura

Si può tentare di ridurre l'albero scartando alcune situazioni limite: alla ricerca minimax si può applicare la potatura «alfa-beta», l'eliminazione cioè di quei rami il cui valore non può essere superiore al massimo che risulta finora garantito (alfa) o non può essere inferiore al minimo che l'avversario ci avrebbe già virtualmente inflitto (beta). Per esempio, se una certa mossa, per l'esame sviluppato, ci garantisce la cattura del re, è inutile esaminare ulteriori alternative ad essa; ugualmente, se in una certa situazione l'avversario è in grado di dare scacco matto è inutile esplorare che cosa accade se non lo fa, e conviene piuttosto risalire al livello precedente ed evitare di cacciarci nei pasticci. Ma anche questo non cambia l'ordine di grandezza di un albero tanto mostruoso; quel che ci serve è un sistema di valutazione posizionale più ampio del semplice riconoscimento della cattura del re, e che non si limiti

a ribaltare ricorsivamente il problema appellandosi a sua volta a una ricerca esaustiva di tutti i conseguenti sviluppi. Una più lungimirante valutazione delle mosse potrebbe attribuire un peso alla cattura o alla perdita di ciascun pezzo, alla mobilità, al predominio sul centro, alla difesa del re e così via. Al limite, la funzione ideale dovrebbe fornire l'esatta valutazione semplicemente considerando la configurazione sulla scacchiera, senza alcuna necessità di esplorare i possibili sviluppi; non vi sarebbe alcuna necessità di esaminare un albero di mosse, ma sarebbe sufficiente considerare le mosse di primo livello e scegliere quella col punteggio più alto. (A rigore, una onesta applicazione del minimax non approdrebbe ad alcuna mossa: il sistema dichiarerebbe subito il matto, la patta, oppure... l'abbandono!). Naturalmente, non esiste alcun criterio assoluto, ricavabile dalle regole formali del gioco, per valutare la posizione in campo: si possono soltanto trarre alcuni criteri empirici dalla letteratura scacchistica o dal giudizio degli esperti, combinando una fase di valutazione approssimativa con una esplorazione controllata degli sviluppi più «promet-

tenti». In base alla affidabilità della valutazione, si possono eliminare fin dall'inizio alcune alternative decisamente scadenti e si può applicare con più profitto la potatura alfa-beta; poiché però l'orizzonte di visibilità totale rimane necessariamente limitato, non si può escludere di aver scartato soluzioni alla lunga più efficaci solo perché per il momento appaiono scarsamente promettenti. Detto per inciso, è proprio grazie ad una estrema variabilità dell'orizzonte esplorato, più che a funzioni di valutazione originali, che i programmi per il gioco degli scacchi cominciano a sconfiggere anche maestri internazionali. La grande velocità degli elaboratori attuali, che non avrebbe aumentato in modo significativo l'efficacia di tecniche di ricerca esaustive o comunque standardizzate, viene sfruttata per studiare a fondo le situazioni più critiche, o almeno riconosciute come tali; è quasi certo che per questa via anche il campione mondiale di scacchi, come è già accaduto per il backgammon e per il nim, sarà prima o poi sconfitto da un programma di computer. Per il momento, tuttavia, dopo aver sconfitto nel 1990 la macchina «Deep

Effetto orizzonte

L'effetto di una capacità previsionale limitata può essere meglio visualizzato se immaginiamo il terreno di ricerca di soluzioni sotto forma di superficie ondulata, con varie convessità e avvallamenti. La soluzione del problema consiste nel trovare un percorso che conduca dal fondovalle fino alla sommità del monte; la funzione di valutazione è qui una specie di altimetro che misura l'avvicinamento all'obiettivo. Di solito ci muoveremo lungo la linea di massima pendenza, ma se finiamo in cima ad una collina secondaria (massimo locale) è necessario scendere per un tratto, allontanandoci momentaneamente dallo scopo, nella speranza di tornare prima o poi a risalire. Se però il nostro orizzonte è limitato, al punto che non riusciamo a vedere le cime circostanti, non sappiamo che direzione prendere e possiamo supporre che non vi siano altre vette raggiungibili; ci rassegnamo allora ad abbandonare la partita o a

trincerarci in uno stallo, mentre forse è ancora possibile attraversare la valle e salire dal lato opposto. Nel caso degli scacchi, è fondamentale mantenere un orizzonte quanto più ampio possibile, e quasi mai si può scegliere la linea di massima pendenza, che è una tecnica locale piuttosto imprudente: se la salita è invitante, converrà ampliare al massimo l'analisi degli sviluppi per escludere che si tramuti in una cengia senza uscita.



Thought» Gary Kasparov si dichiara in grado di battere qualsiasi programma: forse perché è sempre possibile inventare «imbuti» abbastanza profondi da attirare una valutazione algoritmica, o forse perché il metodo minimax, il quale assume che l'avversario giochi con gli stessi criteri adottati dal programma, per definizione esclude astuzie, trabocchetti e tattiche psicologiche che risultano decisive negli incontri di massimo livello.

L'intelligenza alla Sherlock Holmes

I metodi di ricerca di soluzioni e l'individuazione di cammini ottimali lungo alberi o grafi non si applicano soltanto alla teoria dei giochi, ma a qualunque problema sufficientemente formalizzato. Diversi testi classici sull'ia assumono anzi questa prospettiva come fondamentale punto di approccio all'intelligenza artificiale, riconducendo anche aree apparentemente lontane ad applicazione specialistica di tecniche generali di ricerca; si veda tra gli altri Elaine Rich (Intelligenza artificiale, McGraw-Hill 1986) e Bertram Raphael (Il computer che pensa, Muzzio 1986). C'è però il rischio di fornire una visione unilaterale del concetto di intelligenza, eccessivamente rigido e logicista. Il processo razionale non equivale necessariamente ad un dipanarsi serrato di deduzioni e controdeduzioni, come nelle dimostrazioni di teoremi o nelle ardite supposizioni di Sherlock Holmes. Navigare su e giù lungo l'albero delle possibilità e operare previsioni rigorose presuppone non solo un rigido determinismo degli eventi possibili, ma anche la completa conoscenza delle leggi implicite: il che raramente può applicarsi al di fuori di contesti artificiali, immaginari o fortemente semplificati. Una rappresentazione eccessivamente schematica dei problemi è votata al fallimento al momento della applicazione pratica, ma soprattutto è l'antitesi di una più profonda comprensione del mondo reale. L'intelligenza non può prescindere dalla logica e dal ragionamento ipotetico-deduttivo, ma l'uso stesso delle regole formali richiede una capacità di giudizio non riducibile a

sua volta a meta-regole espresse nel medesimo impianto formale.

Macchine astratte

Un secondo modo di descrivere l'evoluzione dell'ia passa attraverso le nozioni di linguaggio orientato ai problemi e di macchina astratta. L'idea di von Neuman di considerare i programmi come dati particolari da memorizzare nel computer, da riutilizzare in modo parametrico e rilocabile nelle più diverse occasioni, e infine da rielaborare automaticamente per adattarli a compiti più generali, ha aperto la via a un processo illimitato, alla creazione di mondi software sempre più complessi e versatili. Salvo considerazioni di efficienza e di costo, un progettista impe-

gnato a produrre software di alta qualità non è condizionato dai linguaggi e dalla operatività della macchina fisica e può descrivere i problemi nel modo più consoni al suo tipo di approccio, specificando le funzioni e gli schemi che gli risultano necessari. Il Lisp, che è il linguaggio maggiormente diffuso nella comunità dell'ia a partire dalla sua definizione negli anni '60 ad opera di McCarthy, esalta la possibilità di costruire funzioni sempre più libere dai vincoli del già dato consentendo di esprimere nello stesso formalismo sia le premesse che le operazioni da effettuare su di esse. L'interprete Lisp può costruire e modificare parti di programma per poi eseguirle nel corso della medesima elaborazione, fino a risolvere lungo il cammino problemi di livello

La visita a... un albero binario

Supponiamo di aver costruito un albero binario contenente una lista di numeri ordinati in modo crescente (utilizzare una struttura ad albero binario è una buona strategia di sort). Ciascuna «biforcazione» dell'albero contiene un numero; il ramo di sinistra porta numeri ad esso inferiori, il ramo di destra porta numeri più grandi. Il «bilanciamento» dell'albero e la sua particolare configurazione è per ipotesi casuale e dipende dalla sequenza originale con cui i numeri sono stati inseriti. A questo punto, parrebbe che il compito di listare i numeri in modo ordinato debba tener conto di tutte le possibilità in cui l'albero si è strutturato. Partendo dalla radice, la «visita» deve operare «prendendo ogni volta il ramo di sinistra» (finché ce n'è), stampando infine il numero terminale, che è il numero inferiore della lista. Se esso ha un ramo di destra, bisogna arrivare al suo primo nodo e poi nuovamente «prendere a sinistra» (come sopra); quando un nodo non ha il ramo di destra, occorre ridiscendere fino ad una biforcazione non ancora stampata, stamparne il numero, e poi ripetere il procedimento sull'eventuale ramo destro... Chi non è molto esperto, finisce per costellare le biforcazioni di segnali indicatori e rischia comunque di tralasciare fronde intere, o

di restare per sempre a mezz'aria: provare per credere. Una procedura ricorsiva è invece anzitutto estremamente chiara. Listare un albero binario equivale a: Se l'albero non è nullo, Listare il ramo di sinistra, stampare il numero del nodo, Listare il ramo di destra. Punto e basta. Tanto «il ramo di sinistra» che «il ramo di destra» sono alberi binari, per cui quella parte del compito è svolto... dalla medesima procedura che abbiamo definito. Parrebbe che la parte più difficile sia solo rimandata al passo successivo, e invece la sorpresa è che non vi è una parte difficile, la complicazione sorgeva solo dal controllo del percorso e per il resto si tratta solo di stampare dei numeri. La regressione all'infinito è evitata dall'elegante (e lapalissiana) annotazione che solo se l'albero non è nullo occorre fare qualcosa.



secondario che in un programma tradizionale occorre invece affrontare a priori.

Il metodo tipico per realizzare processi incredibilmente articolati, più di quanto si potrebbe concretamente ottenere se si dovessero precisare tutti i passi necessari, è l'appello alla ricorsione. In matematica, una definizione ricorsiva permette di costruire un qualsiasi elemento di una sequenza sulla base dell'elemento precedente, posto che sia definito il primo elemento della serie stessa. Dietro questa apparente banalità si nasconde un mondo tanto ricco da indurre il filosofo e matematico Henri Poincaré ad affermare che la ricorsione, o induzione matematica, costituisce il punto nevralgico della creatività del matematico e forse della stessa mente umana. In termini di senso comune, una impostazione ricorsiva suggerisce che si è in grado di risolvere un problema, posto che siano risolti tutti i sottoproblemi implicati dalla soluzione individuata. Il punto è rappresentare il processo risolutivo come una serie di passi, descrivendo un generico passo N a partire da quanto realizzato al passo N-1. (A volte, come nell'esempio illustrato nel riquadro, non occorre nemmeno indicare esplicitamente la sequenza dei passi). In ogni caso, la semplicità della formulazione consente di mettere a fuoco la struttura del problema più che le infinite situazioni in cui può presentarsi. L'autore che più diffusamente ha adottato il tema della ricorsione come chiave di lettura della ricerca in Ia è Douglas R. Hofstadter nel suo affascinante «Goedel, Escher, Bach» (1979, trad.it. Adelphi 1984). Il paradosso dell'autoriferimento, difficilmente dominabile nell'ambito della logica classica, non è che un aspetto del rispecchiamento strutturale tra i diversi piani della realtà, tra, in ultima analisi, la nostra mente e il mondo che tentiamo di decifrare. Hofstadter sembra suggerire che la linea di confine tra il pensiero profondo e l'assurdo è determinata più dai nostri strumenti concettuali che da barriere assolute. L'approccio ricorsivo, a cui non siamo avvezzi, ma che nella stessa programmazione di computer promette risultati

persino difficili da concettualizzare, potrebbe aggirare anche quelli che a priori appaiono come ostacoli di principio alla riproduzione dei processi cognitivi.

Anche la fiducia nello strumento della ricorsione e nei salti di livello concettuale tra i diversi ordini di grandezza e di complessità si muove in un ambito squisitamente linguistico. Il linguaggio può essere la chiave della conoscenza e del pensiero, ma voler ridurre ogni problema ad una questione linguistica ed espressiva potrebbe risultare una forzatura, o almeno una visione troppo parziale. Nell'esperienza cosciente non c'è solo l'aspetto della elaborazione concettuale, ma anche quello della immediatezza sensibile, che appare più un punto di partenza che non un prodotto limite di funzioni complesse.

Regole di produzione

Uno dei primi programmi di intelligenza artificiale, e certamente quello che al congresso di Dartmouth del '56 (al quale si fa risalire l'atto di nascita dell'Ia) impressionò di più i partecipanti, è stato il Logic Theorist di Newell, Shaw e Simon. L'idea di base del progetto, poi perfezionato nell'ambizioso Gps (General Problem Solver), è la cooperazione di più processi produttivi che agiscono ogni qualvolta riscontrano le condizioni adatte al proprio intervento. Nata da teorie sociologiche ed economiche, questa prospettiva non fa appello ad un piano generale, ad una supervisione finalizzata una volta per tutte; in linea di principio, i singoli elementi vivono di vita propria e, pur interagendo tra di loro, hanno ciascuno scopi particolari e proprie soglie di azione. La forma più semplice è data da regole di deduzione da alcune premesse a ulteriori conclusioni; tali capacità deduttive intervengono continuamente, tentando di ampliare e generalizzare i fatti finora noti. A loro volta, le conclusioni fungono da premesse per altre regole, fino a che vi è materia per ulteriori deduzioni «interessanti». Questo schema può essere perfezionato all'infinito, per esempio organizzando (o modificando) la priorità tra i vari

processi, attribuendo gradi di probabilità a ciascuna regola, complicando la struttura dei fatti di base su cui le deduzioni si applicano, e così via. Il punto cruciale è dato dal controllo della proliferazione di nuove «conoscenze»: quando ci si muove in direzioni «interessanti», e quando invece si producono combinazioni astruse e prive di ogni utilità? Come indirizzare (e misurare) l'approssimarsi alla soluzione del compito assegnato al sistema? In primo luogo, si possono marcare alcune delle conclusioni come fondamentali, in modo da arrestare il processo quando queste (o la loro negazione) vengono infine raggiunte; ma anche così il procedere in avanti (forward) può disperdersi in mille direzioni e non imbattersi mai, in tempi ragionevoli, nel risultato atteso. Si può allora applicare il metodo all'indietro (backward), partendo cioè dai fini e privilegiando quelle regole che potrebbero produrli, se le loro premesse venissero soddisfatte. Tali premesse divengono quindi fini secondari, che a loro volta si cerca di soddisfare; quando il processo arriva a pescare nei dati di fatto acquisiti, inizia a risalire la catena «produttiva» fino ad ottenere l'obiettivo di partenza. Come si vede, il «forward» riecheggia la problematica delle strategie di ricerca di soluzioni, mentre il «backward» fa più esplicitamente appello a strategie ricorsive; l'accento è però spostato sulla relativa indipendenza delle varie «regole», che possono sovrapporsi l'una all'altra o perfino competere in tesi contraddittorie. La «robustezza» del sistema è garantita dalla eventuale ridondanza di processi, ma soprattutto dalla possibilità di sopravvivere a conflitti locali o a risultati parzialmente contraddittori. Vi è insomma una maggiore aderenza all'aspetto pratico, per cui in linea di principio la stessa casualità delle interazioni segna una direzione temporale termodinamica e un limite di ciò che concretamente può essere realizzato. Mentre la ricerca combinatoria rischia di impegnarsi in generazioni senza fine e il procedimento ricorsivo non produce alcun esito fino a quando l'intero processo non restituisce, attraverso la catena di autoriferi-

menti, il risultato finale, un sistema di produzione ricava continuamente alcuni risultati parziali e in ciascuna fase può dare conto del lavoro svolto — il lavoro che fino a quel punto poteva essere fatto.

Sistemi fisici di simboli

Il sistema di produzione in definitiva riduce la drammaticità dell'aspetto formale, del come le cose vanno fatte, e punta l'attenzione su che cosa produrre (e che cosa considerare conoscenza di partenza). La ricerca successiva si è sbizzarrita a strutturare gli elementi su cui lavora un sistema di produzione. Roger Shank, interessato a processi linguistici, ha chiamato «script» il dato di base dei suoi programmi, i quali sono in grado di riassumere brevi racconti e riorganizzare la conoscenza in modo da rispondere a domande circostanziate. Marvin Minsky ha introdotto la nozione di «frame», uno schema concettuale che riporta gli attributi fondamentali dell'ente rappresentato deducendoli per default, a meno di specifiche individuali, dal frame di livello superiore, il quale descrive la classe di appartenenza. In ogni caso, quando Newell e Simon parlano del loro «sistema fisico di simboli» il riferimento al mondo reale è demandato appunto all'aspetto simbolico degli elementi elaborati, al loro significato specifico; è per lo meno strano che i critici continuamente a ribadire che, in quanto sistema di elaborazione, si resta nell'ambito dei sistemi formali e con la sola sintassi non si può mai costruire alcunché (tantomeno riprodurre il pensiero intelligente). Se si prescinde un attimo dal formalismo assiomatico della logica moderna, si riconoscerà che le relazioni su cui si esercita il ragionamento intelligente vogliono essere relazioni tra cose, o almeno tra le rappresentazioni che ce ne facciamo: non è un gioco vuoto e gratuito, se gli elementi trattati significano qualcosa. Un «sistema fisico di simboli» è teso a significare qualcosa; quando definiamo le procedure di controllo ovviamente consideriamo le relazioni formali, ma nella applicazione concreta i simboli sono interpre-

tati e intendono riferirsi ad oggetti ed eventi del mondo. Si aggiunga che, in quanto «fisico», un sistema di elaborazione è esso stesso un oggetto che interagisce col mondo reale. Nel momento in cui un insieme di regole formali viene interpretato da un programma e agisce in una macchina concreta, esce dal limbo delle astrazioni sintattiche ed entra in relazione col proprio ambiente. L'insieme di regole diviene in grado di operare, di fare qualcosa — al limite, qualcosa di diverso da quello che il progettista si aspettava... È difficile sostenere che tutto questo resti confinato, per definizione, nel «vuoto formalismo».

Il Prolog

Forma e contenuto, sintassi e semantica, sono i poli di una distinzione concettuale che si può applicare con maggiore o minore enfasi; nemmeno la programmazione a computer si può ridurre ad uno solo di essi. Tuttavia alcune direzioni di ricerca hanno finito per privilegiare l'uno piuttosto che l'altro aspetto dei programmi di Ia. È il caso della programmazione logica, che dell'impianto concettuale delle regole di produzione ha enucleato la problematica logico-formale. Nel sistema di produzione non vi è una rigida gerarchia di funzioni: ciascuna legge logica interviene «appena può», appena verifica la sussistenza delle condizioni a cui è vincolata. Il linguaggio definito da Alain Colmerauer nel '72, il Prolog (nella versione originale di Marsiglia, perfezionata da Kowalski, e in quella ormai classica di Edimburgo), è basato su un interprete che esamina sistematicamente gli asserti del «programma» nell'intento di ricavare la dimostrazione del risultato richiesto. Secondo le «equazioni metaforiche» proposte da Sterling e Shapiro (The art of Prolog, Mit Press 1986), in Prolog il programma equivale ad un insieme di assiomi e l'esecuzione equivale ad una prova costruttiva, ricavata dal programma, di un asserto-meta (goal). Nonostante i problemi di fondo legati alla prospettiva costruttivista (la questione della negazione come fallimento, l'ipotesi del

«mondo chiuso», la decisione sulla finitezza della ricerca ricorsiva) e alla tensione tra purezza logica e contaminazioni pratiche (per esempio, in Prolog le funzioni di input e di output sono ottenute come «effetti collaterali» di verifiche scontate in partenza), la programmazione logica è diventata il cavallo di battaglia dell'Ia europea e ha costituito l'ossatura per la scommessa della quinta generazione giapponese. Come nel Lisp, un processo Prolog è ampiamente ricorsivo; la formulazione dei fatti e la definizione del problema risulta perciò compatta e suggestiva. A differenza del Lisp, invece, un programma Prolog non è procedurale: stabilisce che cosa deve essere fatto, non come e in quale sequenza, compito totalmente demandato all'interprete e, a parte gli effetti collaterali, di per sé inessenziale. Questa distinzione di principio tra il che cosa e il come è un passo molto importante, che segna una soluzione di continuità con la concezione dei fini e dei mezzi, delle procedure di alto e di basso livello, che caratterizzava il top-down e la programmazione strutturata. L'affermazione di Lady Lovelace, «la macchina può fare solo quello che le si dice di fare», resta in un certo senso vera, ma comincia ad assumere sfumature più interessanti. Naturalmente, l'interprete Prolog si muove sul puro terreno linguistico, il suo «che cosa» ha connotazioni esclusivamente sintattiche; la programmazione logica ha immediate conseguenze operative (per esempio, si sposa perfettamente con l'elaborazione in parallelo, visto che ciascuna clausola e ciascun sottoproblema può essere esaminato in modo indipendente e atemporale), ma non è sufficiente per procedere verso una migliore simulazione degli aspetti semantici dei problemi reali. Il successo del Prolog, a giudicare dal suo rapido diffondersi in Italia negli ultimi anni '80, è da ascrivere alla sua eleganza formale e alla rispettabilità teorica dei suoi principi di fondo, che presentano numerosi punti di contatto con le discipline logico matematiche classiche e lo hanno reso ben accetto al mondo accademico. Data la vocazione interdisciplinare dell'Ia, ciò è risultato

di grande aiuto, perché ha comportato l'irruzione di forze fresche e spesso autorevoli, spingendo matematici e logici a confrontarsi col tema del ragionamento artificiale e della riproduzione dei processi deduttivi non soltanto nei mondi astratti dei sistemi assiomatici, ma anche nella realtà bizzosa dei programmi di computer. La profusione di Prolog ha consentito una rilettura (e una riscrittura) di tutta la problematica dell'la classica, comprese le strategie per la soluzione dei problemi (con interpreti Prolog via via più raffinati e tecniche di controllo più sofisticate), lo sviluppo di macchine astratte (con meta-interpreti e ambienti linguistici di ordine superiore) e la riproduzione di strutture dichiarative vicine al linguaggio naturale e alla sua grammatica.

Si è però creata una situazione vagamente schizofrenica. Da un lato, del Prolog si è ulteriormente esasperato il versante astratto e formale, con molta modellizzazione di teoremi e poca versatilità pratica; ed anche così il logico puro si dichiara insoddisfatto per le «contaminazioni» recate dagli effetti collaterali e dalle esigenze di programmazione sulla macchina, oltre che scettico circa la possibilità di meta-interprete in meta-interprete, di schivare le aporie logiche indicate dai teoremi di Goedel. In parole povere, il Prolog non sarebbe abbastanza «puro»; e se poi lo fosse, come potrebbe essere consistente e completo un sistema formale abbastanza ricco da meta-interpretarsi e valutare le proprie stesse strategie deduttive? D'altro canto, per le applicazioni pratiche il Prolog è troppo esigente e diviene necessario costruire meta-interpreti più tolleranti, per i quali naturalmente i teoremi di finitezza o di consistenza non valgono più e che operano del resto in limitati contesti applicativi. Il Prolog diviene così un semplice linguaggio di implementazione, comodo ma poco efficiente; la sua efficacia dipende dalla strategia adottata dal meta-interprete o dalla normativa stabilita nel nuovo ambiente di sviluppo. In altri termini, benché il Prolog sia già una specie di sistema esperto a regole di produzione (con l'interprete in veste di motore inferenziale e le righe di codice

come asserti e regole deduttive), per avere un sistema esperto ragionevolmente operante è opportuno ridefinire sia le strategie dell'interprete, sia la grammatica delle regole e la loro organizzazione.

Marvin Minsky e la società della mente

Le accentuazioni semantiche e applicative dei programmi di Ia hanno trovato invece piena accoglienza in altri gruppi di lavoro, per esempio nella direzione indicata da Minsky nel suo «La società della mente» (1985, trad. it. Adelphi 1989), un complesso affresco dei problemi e delle possibili soluzioni della ricerca in Ia. Secondo Minsky, la mente non è qualcosa di individuale e di semplice, ma è una stratificazione di più «menti» tra loro in competizione, è una società dinamicamente organizzata in una infinità di livelli operativi, unità ricettive, centri decisionali, elementi di controllo, di memorizzazione, di inibizione. È come un assemblaggio di centinaia di macchine radicalmente diverse, ciascuna operante in maniere differenti, tra cui forse alcune modalità riproducibili mediante programmi di computer. I concetti sono individuati e interconnessi dall'incessante lavoro di un'immane burocrazia di «agenzie» e di agenti, ciascuno dedicato ad un particolare aspetto e ad una particolare funzione. Questa interpretazione, che certamente non si colloca al livello dei singoli neuroni e delle infinite sinapsi cerebrali, tenta di interpretare le funzioni superiori in termini di funzioni più semplici (e di uso più generale), trattando di idee e di nozioni di origine psicologica, pedagogica e introspettiva. Essa costituisce per Minsky anche un possibile progetto di mente artificiale, da valutare e da mettere a punto tramite la sperimentazione di sistemi e programmi adeguati. La ricostruzione di Minsky non vuole essere la copia esatta della mente umana, quanto piuttosto un modello concettuale, un precipitato di tutte le teorie disponibili, di tutto quello che sappiamo in fatto di conoscenza, coscienza e mondo mentale. Più le nostre teorie psicologiche e neu-

ambizioni aziendali

La vostra azienda è ambiziosa.

Esige il massimo, per vantare efficienza ed immagine.

Uno sguardo ai sorprendenti

prodotti **Genel** è d'obbligo.

Voxmail ad esempio, integrato nel Vostro centralino, realizza un servizio di posta elettronica vocale centralizzata con distribuzione automatica dei messaggi. Riservato, facile da usare, intelligente.

Visiman, che riceve il visitatore, lo identifica, emette un passi, avverte del suo arrivo e lo guida all'interno dell'azienda.

Autodial, sistema che snellisce il traffico telefonico con funzioni in sintesi vocale direttamente rivolte a centralinisti non-vedenti. Incredibile vero?

E poi **Master**, **Superlink**, **Statel**, sistemi integrabili, performance ai confini del possibile telematico ad un costo rassicurante.

Assecondare le ambizioni è una scelta che rende. Parliamone.

Genel

Generale Elettronica S.r.l.
10142 Roma - Via Mellotroni Marzola, 80
Tel. (06) 244960 - 242543 - 5191061 Fax (06) 2611987
Via Persico, 118 - 02141 Milano
Tel. (02) 2674282 - 2674285 - 2674286 Fax (02) 2674284

3

Gruppo 31 - Industria Italiana Informatica

ACE ROMA

rologiche saranno affidabili, tanto più il modello artificiale saprà funzionare e fornire indicazioni ulteriori. È importante quindi disporre di teorie — molte teorie — sui meccanismi cognitivi individuati; nella modellizzazione non si pretende di copiare esattamente il processo naturale, ma ci si accontenta di una riproduzione funzionale, al livello di analisi a cui la teoria fa riferimento. Con la nota metafora di McCarthy, si ripete che la ruota non riproduce fedelmente la locomozione animale, eppure per lo più svolge egregiamente il suo compito, spesso assai meglio del sistema naturale.

Cognitivismo

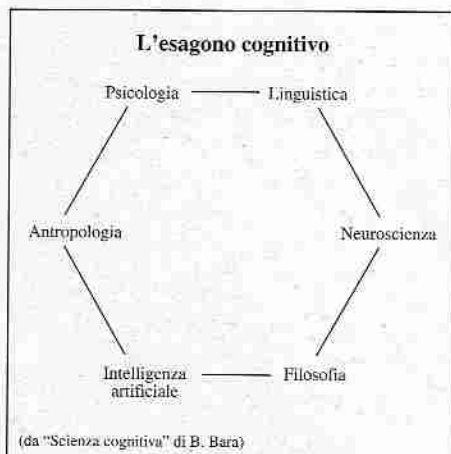
Pur con sfumature differenti, l'Ia classica è dunque caratterizzata da un approccio simbolizzante: i programmi trattano entità e dati che, per decisione esplicita del programmatore, simboleggiano oggetti, eventi, relazioni e concetti. In parte, questo sembra accadere anche in una procedura gestionale, nella quale determinate variabili e celle di memoria contengono valori che rappresentano prezzi, giacenze, tipi di articoli, lettere di amministrazione, nominativi. Da un punto di vista gestionale, però, la relazione con l'oggetto rappresentato è quasi soltanto una faccenda di input e di output, di titoli e decodifiche sugli stampati e di appropriate maschere sul video. Per una procedura di magazzino, bulloni e palle da tennis sono esattamente la stessa cosa. Anche in un moderno data base, l'obiettivo della equivalenza strutturale fra entità e relazioni da una parte e mondo reale dall'altra, tra vista logica e realtà aziendale, trae origine più da una questione di opportunità («robustezza» dello schema concettuale, unicità dei dati, condivisione delle informazioni tra procedure concorrenti) che non da un preciso intento di rappresentazione simbolica. Persino in un programma di simulazione si tende ad isolare alcune quantità astratte e a riprodurre il loro andamento complessivo senza alcun interesse o alcuna pretesa di riportare sulla macchina il nostro modo di concepire il fenomeno raffigurato. L'Ia si

ripropone invece di rappresentare fatti e relazioni proprio allo scopo di trarne conseguenze e decisioni analoghe a quelle ricavate da una mente intelligente. Non solo l'elemento simbolico deve avere un rapporto molto più ampio con l'oggetto simbolizzato, ma è in relazione anche col concetto che noi ne abbiamo, con l'intera rete concettuale in cui è immerso. Si tratta insomma di un esplicito rapporto a tre fra simbolo, oggetto e concetto psicologico, ovvero tra programma, mondo e mente. È vero che tutto questo spesso rimane velleitario, dal momento che sappiamo ben poco che cosa possa essere un «concetto» o come funzioni la mente, che del mondo abbiamo un'idea approssimata e soggettiva, e che persino i nostri programmi ci sfuggono di mano e si mettono a fare cose del tutto inaspettate (e in genere poco interessanti). Eppure l'eventuale ingenuità di certe emulazioni non fa che rispecchiare la povertà della corrispondente teoria psicologica, riflettendo lo stato delle nostre conoscenze: non è un difetto del programma, quanto piuttosto un limite della ricerca nel settore. Si comprende così come l'Ia abbia attinto da ogni possibile disciplina collaterale e soprattutto abbia fortemente promosso ulteriori ricerche e teorizzazioni. Lo status interdisciplinare non dipende da una difficoltà di collocazione, quanto piuttosto dal reale interscambio con la neuroscienza, la psicologia della mente o l'epistemologia. Sotto questo profilo, i tentativi dell'intelligenza artificiale costituiscono per le scienze della mente un prezioso banco di prova, in quanto permettono di sperimentare 'in vitro' l'efficacia di ogni teorizzazione, fino a incarnare interamente una nuova teoria e a farla agire con le proprie forze e i propri schemi interpretativi in situazioni realistiche — beninteso, per quanto si spinge la simulazione a computer. Alcuni ricercatori epistemologicamente più attenti e meno ancorati a schemi comportamentali tradizionali hanno tentato di interagire più attivamente con le proposte (o le promesse) dell'Ia. Lo psicologo Samuel Papert, per esempio, si è impegnato sul terreno stesso

dell'intelligenza artificiale, collaborando con Minsky ed evidenziando la centralità del processo di apprendimento e in generale degli aspetti evolutivi dell'attività cerebrale. Nello stesso tempo, ha proposto un linguaggio di programmazione espressamente pensato per stimolare l'intuizione logica e topologica del bambino: il Logo, primo esempio di software didattico fondato su precise ed esplicite istanze pedagogiche.

La scienza cognitiva

La matura consapevolezza che l'impatto dell'Ia sulle altre discipline stava suggerendo una nuova prospettiva scientifica, può esser fatta risalire alla prima conferenza della Cognitive Science Society, tenuta nell'agosto 1979 alla University of California di San Diego. Secondo lo psicologo Bruno Bara (Scienza Cognitiva, Boringhieri 1990, pag.44), «la conferenza fu un successo internazionale, e rimane nei miei ricordi personali come un'incredibile sequenza di relazioni e discussioni una più eccitante e dirompente dell'altra». Tra i partecipanti, Bara cita il linguista Lakoff, il filosofo Searle, gli psicologi Johnson Laird, Norman e Rumelhart, il neuroscienziato Geschwind, gli studiosi di Ia Minsky, Newell, Schank, Simon e Winograd. In sostanza, il carattere distintivo della scienza cognitiva è l'accettazione della metodologia simulativa nello studio della mente umana. Intesa come linea di demarcazione, la possibilità di implementare programmi che simulano processi psichici o neurali non è un semplice strumento aggiuntivo, ma costituisce un salto di qualità tale da giustificare una nuova definizione della prassi scientifica. Per lo scienziato cognitivo, una teoria della mente è scientifica solo se è possibile costruire un modello che esibisca il comportamento previsto; inoltre, il modello deve essere traducibile in un programma di Ia. Il successo o l'insuccesso della simulazione funzionano alla stregua di esperimenti di laboratorio, consentendo di corroborare o falsificare la teoria e soprattutto suggerendo ulteriori revisioni e approfondimenti. Infatti «la costru-



zione di un programma di calcolo è una tale sfida intellettuale, un tale evidenziatore di debolezze, da spingere gli studiosi a continue revisioni e approfondimenti della teoria di partenza». In negativo: «se sulla base di una teoria data non è possibile generare un programma per calcolatore che riproduca gli aspetti ritenuti essenziali della teoria stessa, allora la teoria non è scientificamente accettabile» (Scienza Cognitiva, pag.73). Perciò, anche se dell'«esagono cognitivo» fanno parte psicologia, filosofia, linguistica, antropologia, neuroscienza e intelligenza artificiale, è proprio quest'ultima ad essere chiamata più direttamente in causa e a costituire il referente essenziale della ricerca.

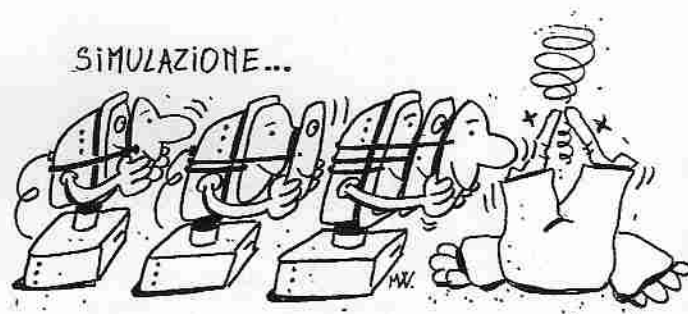
La scienza cognitiva impone però una sua interpretazione dell'Ia, che non necessariamente corrisponde a tutto ciò che l'Ia è o è stata, anche limitandoci alla sua componente cognitivista. L'intento prioritario di studiare e capire la mente umana pone condizioni molto restrittive alle simulazioni richieste ai programmi di Ia; per esempio, le simulazioni dovranno riprodurre non solo le prestazioni positive, ma anche le esitazioni, gli errori e le discontinuità della mente biologica. Più esattamente, nel riprodurre le manifestazioni e i risultati finali di una prestazione intelligente occorre anche rispettare la corrispondenza con i relativi processi mentali di una persona impegnata nel medesimo compito. Questo requisito è molto forte, e forse è addirittura impossibile da verificare: l'equivalenza

delle procedure può essere stabilita solo induttivamente, a partire dagli output e dai risultati; inoltre la teoria, il modello e infine il programma di simulazione introducono livelli successivi di astrazioni e semplificazioni che rendono convenzionale la decisione su che cosa in quel momento è essenziale e che cosa non lo è. In ogni caso, è discutibile che queste limitazioni debbano applicarsi all'intera ricerca in Ia, ad esempio per contraddistinguere una Ia buona («morbida» o «debole» che sia) rispetto ad una Ia destinata invece a mancare i propri obiettivi. Quando il baricentro si sposta dall'esigenza di studiare la mente umana all'idea regolativa della riproduzione in generale di prestazioni intelligenti, un eccessivo antropocentrismo potrebbe addirittura risultare miope.

Una Ia per ogni gusto

Anche da queste poche note si vede come il tentativo di raccontare e circoscrivere l'intelligenza artificiale riconduca ad una molteplicità di storie e di definizioni. Ciascuna prospettiva invita ad una differente lettura degli eventi determinanti, ad una diversa disposizione delle caselle e dei punti critici. Questo può sconcertare chi è abituato a vedere la scienza attraverso il filtro dei manuali di scuola e non si accorge che la realtà della ricerca è tutt'altra cosa. Forse nel caso dell'Ia è più evidente quello che altre discipline dissimulano meglio, cioè che l'oggetto dell'indagine è sempre ipotetico e sfuggente, e ad ogni approfondimento si colora di connotazioni imprevedute. La ricchezza di idee e di prospettive testimonia anzi dello stato di salute del settore, che si è conquistato uno spazio di grande rilievo. Non è più possibile costruire paradigmi psicologici o neurologici senza tener conto dei problemi impostati dalla programmazione a computer, né implementare un model-

lo decisionale senza giovare della vasta esperienza oggi acquisita. In tale fase di rapido progresso, è difficile però prevedere la direzione dei passi successivi, perché la meta — sia essa un modello generale della mente, la riproduzione di capacità razionali o la costruzione di nuove forme di «intelligenza» — è troppo vaga e problematica per costituire un punto fisso su cui regolare gli strumenti di viaggio. Come Colombo sulla via delle Indie, possiamo solo congetturare quello che l'orizzonte ci nasconde; la discussione sulle tappe compiute e sulla strada che resta è cer-



tamente utile, se non altro per rincuorare l'equipaggio, ma in questo caso il vero fine è il viaggio stesso. Tra gli imprevedibili ricicli e rifondazioni che incrociano il cammino dell'Ia, vedremo la volta prossima le ragioni di una impostazione anti-cognitivista: il connessionismo neurale. ■

Luciano Bazzocchi

(2 - continua)

BIBLIOGRAFIA

- Bruno Bara, *Scienza cognitiva*, Boringhieri 1990.
- Jeremy Bernstein, *Uomini e macchine intelligenti*, Adelphi 1990.
- John Haugeland, *Intelligenza artificiale*, Boringhieri 1988.
- Douglas R. Hofstadter, *Goedel, Escher, Bach*, Adelphi 1984.
- Marvin Minsky, *La società della mente*, Adelphi 1989.
- Marvin Minsky (a cura di), *La robotica*, Longanesi 1987.
- Roger C. Schank, *Memoria dinamica*, Marsilio 1987.
- Leon Sterling, Ehud Shapiro, *The Art of Prolog*, Mit Press 1986.

Specializzato
Formazione oggi (3ª parte)
Quale futuro per la
pubblica amministrazione?

l'organizzazione

l'automazione e le comunicazioni dell'ufficio

office[®] automation

EDIZIONE INTERNAZIONALE - 20124 MILANO, VIA SETTELA 5 - MONDO - ANNO UNDICESIMO - N. 01 OTTOBRE 1991 - SPED. IN ABB. POST. GR. 8/73

Oggi con NCR DMS,
tutte le vostre informazioni su carta
possono finalmente integrarsi
con quelle al computer.



NCR

Open, Cooperative Computing.
La scelta strategica per i grandi cambiamenti.