TO: BGB ✓
RSE ✓
Sri ✓
return to EAF

Machines and the Theory of Intelligence

DONALD MICHIE

Department of Machine Intelligence, University of Edinburgh

This article is based on a series of special lectures delivered at University College, London, in November 1972.

THE birth of the subject generally referred to as "artificial intelligence" has been dated[1] from Turing's paper[2] *Intelligent Machinery* written in 1947. After twenty-five years of fitful growth it is becoming evident that the new subject is here to stay.

The scientific goal of research work in artificial intelligence is the development of a systematic theory of intelligent processes, wherever they may be found; thus the term "artificial intelligence" is not an entirely happy one. The bias towards artefacts is reminiscent of aerodynamics, which most people associate with aeroplanes rather than with birds (yet fruitful ornithological application has been achieved[3]). Here I shall review briefly some of the experimental knowledge systems which have been developed, and indicate how pieces of theory abstracted from these might fit together.

Some Performance Systems

Game playing was an early domain of interest, and Shannon[4], Turing[5], and Newell, Shaw and Simon[6] contributed classic analyses of how machines might be programmed to play chess. The first significant performance system was Samuel's program[7] for checkers, which eventually learned to play at the level of a good county player, far higher than that of Samuel himself. This last circumstance played a valuable part in discrediting the cruder manifestations of the doctrine that "you only get out what you put in".

The fundamental mechanism underlying all this work has been a cycle of processes: lookahead, evaluation and minimaxing. These derive ultimately from a method used to establish a "foregone conclusion theorem" for such games (two person, zero sum, perfect information, no chance moves) which states that the outcome value can be computed on the assumption that both players follow a (computable) best strategy. For a trivial game, such as that schematized in Fig. 1*a*, the computation can actually be performed: all terminal board positions are assigned values by the rules of the game, and these are "backed up" by the minimax assumption that White will always choose the immediately accessible position which has the maximum value and that Black will select the one with the minimum value. Clearly the procedure not only demonstrates a theorem but also defines a strategy.

But what is to be done when, as in any serious game, it is not practicable to look ahead to the end? Turing and Shannon independently suggested looking ahead as far as practicable, to what may be termed the "lookahead horizon", assigning some approximate values to the positions on the horizon by an evaluation function, and backing these up by the same minimax rule. The corresponding strategy says "choose that immediate successor which has the highest backed-up value".

This rule has been proved empirically in numerous game-playing programs, but in spite of its intuitive appeal it has never been formally justified. The question is posed diagrammatically in Fig. 1*b*.

Search procedures form part of the armoury of the operations-research man and the computer professional. Stemming from such work as Samuel's, people concerned with game playing and problem solving have implemented mechanisms for guiding the search, first, by forming sub-problems[8] or, second, by making heuristic estimates of distance-to-goal[9]. Various theorems have established conditions under which such techniques can be used without sacrificing the certainty of termination or the optimality of the solution found[10,12,13].

The use of an "evaluation function" to guide the search is a way of smuggling human *ad hoc* knowledge of a problem in through the back door. There is no cause to disdain such a route; it is after all one of the principal channels through which natural intelligences improve their understanding of the world. At the same time automatic methods have been developed for improving the reliability with which problem states are evaluated[11].

Samuel's early work on game learning[7] indicated that seemingly pedestrian mechanisms for the storage and recall

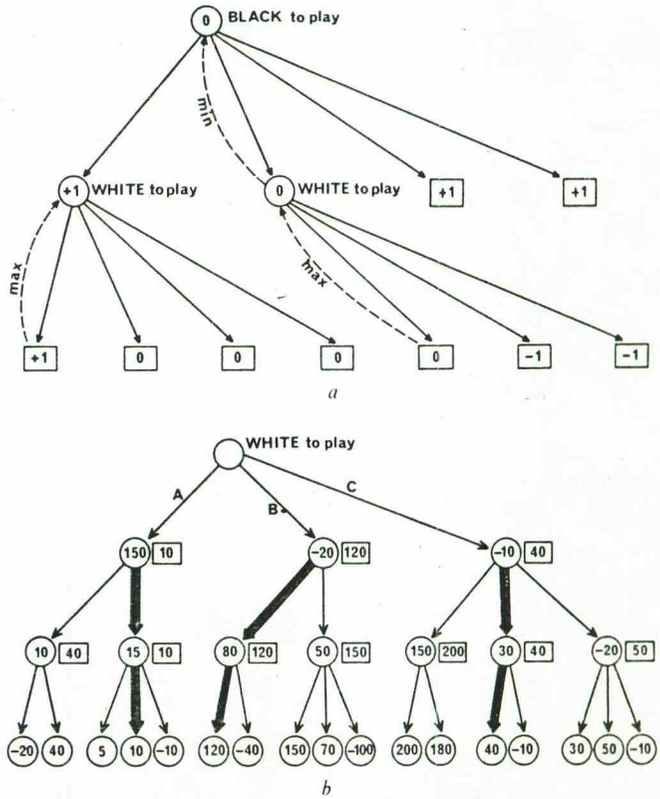

Fig. 1 *a*, The root of this two-level lookahead tree acquires a value by alternate application of the "max" and "min" functions. If alternation is extended backwards from all terminal positions of the game tree, the initial position of the entire game will ultimately be assigned a value. Terminal positions are shown as boxes; *b*, lookahead tree in which the nodes are marked with "face values". Boxed figures are values backed up from the lookahead horizon. If move-selection were decided by face values, then move A would be chosen, but if backed-up values then move B. What is the rationale for B ?

of previously computed results can have powerful effects on performance. Recently the combination of rote learning schemes with heuristic search has been shown to have applications to plan formation in robots[13,14]. To exploit the full power of this combination, whether in game playing, in robotics or in other applications, one would like the rote dictionary to contain generalized descriptions or "concepts" (for example, of classes of game-positions "essentially similar" from a strategic point of view) to be looked up by processes of recognition, rather than by point-by-point matching. Such a dictionary is to be used in the style "If the situation is of type A, then perform action x, if of type B, then action y" and so on. One is then in effect processing a "decision table" which is formally equivalent to a computer program. There is thus a direct link between work on the automatic synthesis of strategies in game playing and robotics, and work directed towards automatic program-writing in general.

Recognition usually involves the matching of descriptions synthesized from sensory input with stored "canonical" descriptions of named objects, board positions, scenes, situations and so on. Choice of representation is crucial. At one extreme, predicate calculus[15] has the merit of generality, and the demerit of intractability for updating and matching descriptions of objects, positions, scenes or situations; at the other extreme lie simple "state vector" representations, which fall down through awkwardness for handling complex interrelationships. Somewhere in the middle lies the use of directed labelled graphs ("relational structures", "semantic nets") in which nodes stand for elements and arcs for relations. Impressive use of these structures has been made in a study of concept formation in the context of machine vision[16].

Language interpretation has been the graveyard of many well-financed projects for "machine translation". The trouble proved to be the assumption that it is not necessary for the machine to "understand" the domain of discourse. One of the first demonstrations of the power of the semantic approach in this area was Bobrow's "STUDENT" program[17] for answering school algebra problems posed in English. A program by Woods, Kaplan and Nash-Webber[18] for the interrogation in English of a data base with a fixed format has been used by NASA scientists to answer questions about Moon rocks. An essay by Winograd[19] on computer handling of English language dialogue, again making intensive use of an internal model of the dialogue's subject matter, has left no doubt that machine translation can only be solved by knowledge-based systems. The knowledge base required to render arbitrary texts non-ambiguous is now recognized to be bounded only by the knowledge possessed by their authors. Winograd compares the following two sentences:

The city councilmen refused to give the women a permit for a demonstration because they feared violence.

The city councilmen refused to give the women a permit for a demonstration because they advocated revolution.

The decision to refer "they" to "councilmen" in the first case and to "women" in the second implies a network of knowledge reaching into almost every corner of social and political life.

Mass spectrogram analysis was proposed by Lederberg as a suitable task for machine intelligence methods. The heuristic DENDRAL[20] program developed by him and Feigenbaum now outperforms post-doctoral chemists in the identification of certain classes of organic compounds. The program is a rich quarrying ground for fundamental mechanisms of intelligence, including the systematic conjecture of hypotheses, heuristic search, rote learning, and deductive and inductive reasoning. I shall refer back to this work later in connexion with the use made by intelligent systems of stored knowledge.

Of all the knowledge systems which have been attempted, robotics is perhaps the most simple in appearance. In reality, however, it is the most complex. The chess amateur can appreciate that Grandmaster chess has depth and subtlety. But there is no such thing as a human amateur at tasks of navigation and "hand-eye" assembly. Every man is a Grandmaster at these tasks, having spent most of his waking life in unwitting but continual practice. Not having been informed that he is a Grandmaster, and having long since stored most of his skill at a subliminal level, he thinks that what seems subjectively simple is objectively so. Experience of research in robotics is a swift and certain cure. Something of the depth of analysis which is required can be gleaned from the discussion by McCarthy and Hayes[21] of the properties which should be possessed by a calculus of situations, actions and causal laws.

The crux of any such calculus is how to represent in a formal language what the robot knows about its world. McCarthy and Hayes distinguish "epistemologically adequate" and "heuristically adequate" representations. (In an earlier generation Ryle[22] contrasted "knowing that" and "knowing how".) "The epistemological part is the representation of the world in such a form that the solution of problems follows from the facts expressed in the representation. The heuristic part is the mechanism that, on the basis of the information, solves the problem and decides what to do."

I shall consider now what is probably the simplest world to be seriously discussed, that of Popplestone's "blind hand" problem (internal report, Department of Machine Intelligence, Edinburgh), with the object of indicating that there is more to robot reasoning than meets the eye, and expanding a little the epistemological-heuristic distinction.

A blind, insentient, robot shares with one or more "things" a world consisting of only two places "here" and "there", and has available to it the actions "pickup", "letgo" and "go". "Pickup" is non-deterministic and causes (if the hand is empty when the action is applied) a "thing" selected at random from the place where the robot is, to acquire the property "held". An initial situation called "now" is defined, in which it is asserted that every thing at "here" (and there is at least one such) has the property "red". A goal situation is defined as one in which at least one red thing is at "there".

Invariant Facts and Laws

The kinds of facts which the robot needs to know include that the robot and anything held by it must be in the same place, and that something cannot be in both places at once. Using a prescription of Green[23], a formalization of this apparently trivial problem in first order logic might start along the following lines. (The variables t, p and s are to be interpreted as standing for objects, places and situations respectively.)

for all t,p,s: held(thing(t),s and at(thing(t),p,s) implies at(robot,p,s),

for all t,p,s: held(thing(t),s) and at(robot,p,s) implies at (thing(t),p,s),

for all p,s: at(robot,p,s) implies at (thing(taken(s)),p,s),

for all t,s: at (t, here,s) implies not at (t, there, s).

The conjunction of these statements describes some of the physics of this world. The last statement, for example, asserts that an object cannot be both at "here" and at "there" in one and the same situation.

The initial situation, "now", is described in like manner:

for all t: at (t, here, now) implies red(t), at(thing (a), here, now).

The latter statement merely asserts that at least one thing (represented by the constant a) is at "here" in situation "now". The function "thing" is a convenience for distinguishing other objects from the robot, whom we may wish to exclude from some otherwise universal statements—like one implying that the robot is "held", for instance.

How can the machine be enabled to reason about the chains of possible consequences derivable from "now" and so to construct an action chain leading to a goal situation? The goal may be defined, using Green's "answer" predicate[24], as:

for all t, s: at(t, there, s) and red(t) implies answer(s).

But how do we handle the actions? The contrast between epistemological and heuristic criteria becomes very sharp at this point. Consider two approaches.

One can go the whole way and stick to formal logic, defining the transition laws of our world under the various actions. For example, the first of the following three "letgo" axioms translates freely: "in the situation produced by doing a 'letgo', nothing is held".

for all t, s: not held (thing(t), do(letgo, s)),

for all t,p,s: at(t,p,s) implies at(t, p, do(letgo, s)),

for all t,p,s: not at(t,p,s) implies not at(t,p,do(letgo, s)),

and similarly for the other actions.

Now the problem of plan construction is reduced to one of logical deduction, in fact deduction of the statement "answer(do(go(there), do(pickup, do(go(here), do(letgo, now)))))". This says, in English, that "the goal situation is the one resulting from doing a "go there" in the situation resulting from doing a "pickup" in the situation resulting from doing a "go here" in the situation resulting from doing a "letgo" in the situation "now" ", and it is clear how this can be re-interpreted as an algorithm.

This deduction can in principle be mechanized, but there are two severe snags. First, the need to incorporate "frame axioms"[24,25] (which spell out all the facts which remain unchanged after the performance of given actions, as in the last logic statement above) escalates for nontrivial problems and renders the automatic deduction process intractable even in the present toy problem. Second, the logic representation is not heuristically adequate.

On the other hand, one can go to the other extreme, and express the whole problem as a computer simulation couched in a suitable programming language, matching situations with data structures and actions with procedures. But this approach encounters difficulties with the epistemological criterion, for the structure of the problem world can be readily complicated so that it can no longer easily be described by the use of simple representations of the "state vector" type. Various attacks are being made on the representation problem in an attempt to make the best of both worlds, the epistemological and the heuristic. Some good early suggestions were made by Popplestone, using essentially the same blind hand problem, and were reviewed in *Nature*[27] two years ago. Since then powerful new programming aids, such as the PLANNER[28], QA4 (ref. 29) and CONNIVER[30] languages have come into play. In addition particular mention should be made of the Stanford Research Institute's study of autonomous plan formation[14,15], in which many of the matters discussed above have been under experimental investigation.

The key ideas on which much work centres is that plan construction should be conceived as a search through a space of states of knowledge to generate a path connecting the initial knowledge state to one which satisfies the goal definition. Everything turns on finding ways of representing knowledge states so that the transformation of one into another can be neatly computed from the definition of the corresponding action ("What will I know about the state of affairs after doing A ?").

Experimental Robotics

The STRIPS system[14,15] at Stanford Research Institute combines reasoning in first-order predicate calculus with heuristic search. In the situation depicted in Fig. 2 the robot must devise a plan for pushing objects around so that one of the boxes end up in room R1, subject to the constraint that at no time must the wedge be in the same room as a box. If the plan goes wrong, the system must be capable of recovering from error state and, if possible, "mending" the failed plan appropriately. Facilities are incorporated whereby successful

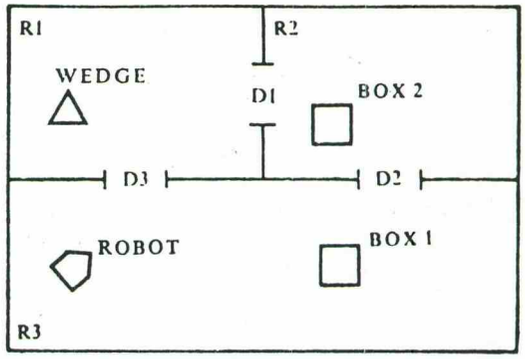

Fig. 2 Robot environment for a constraint problem. (Reproduced from ref. 31.)

plans are automatically "remembered" and their elements recombined for use in appropriate future situations[14].

Following simultaneous development of the idea of optical ranging in Japan[32], Britain (R. J. Popplestone, personal communication) and America[33], Stanford University's robot project uses a laser optical ranging system for mapping the three-dimensional surfaces of "seen" objects. Another branch of the same project is currently able to assemble an automobile water pump comprising two pieces, a gasket and six screws (J. Feldman, personal communication). This is done blind, using mechanical feedback.

At Edinburgh automatic assembly is also under study. Programs exist for packing simple objects onto a confined surface, identifying a limited set of objects by visual appearance, and solving problems of stacking rings on pegs (D. M., A. P. Ambler, H. G. Barrow, R. M. Burstall, R. J. Popplestone, and K. J. Turner, paper to be presented at a Conference on Industrial Robot Technology at the University of Nottingham next month).

In industrial laboratories, notably in America (for example, the Charles Stark Draper Laboratory of MIT) and Japan[34], automatic assembly studies are multiplying.

Idea of a Theory

I have already mentioned the abstracting of pieces of theory from performance systems such as those listed above. What is meant by "theory" in this context ? I have just considered a fragment of simple robot world theory, and one can, of course, speak of a piece of chess end-game theory (for example, that expressed by Tan's program[35] for the two-Kings-and-one-Pawn end-game) or of the theory of mass spectrometry embedded in the heuristic DENDRAL program. One can even, legitimately, speak of Winograd's program as constituting a linguistic theory, or at least as containing or implying one. But these theories are descriptive of specific domains, not of intelligence itself.

It would be naive to pretend that the search for a metatheory is something new, or even that it is anything but old philosophy in new dress. An early name suggested for what is now "artificial intelligence" was "epistemological engineering" (P. M. Woodward, personal communication). The new epistemology, however, has a trick which the old philosophers lacked, namely to express any given theory (of knowledge, reasoning, abstraction, learning and the like) in a sufficiently formal style to program and test it on the machine.

Hence there is no longer a meaningful distinction to be drawn between a theory of some given intelligent function, and an algorithm for carrying it out (which could in turn be converted into a program for some particular machine) together with any useful theorems for describing the algorithm's action. Algorithms, then, are theories, and this has been true for a long time. But there have been no reasonable mechanisms available for handling them. Mathematics, on the other hand, has had the necessary mechanisms for manipulating the formalisms which it uses for describing physical systems. Hence closed-form mathematics has been the "typical" embodiment

of theory in the physical sciences. By contrast, the "typical" embodiment of theory in cognitive engineering is algorithmic.

What Use is Knowledge?

The value of stored knowledge to a problem-solving program again divides into epistemological and heuristic parts. In the first place sufficient knowledge must be present for solutions to be in principle deducible. But that is only the start. Heuristically, the value of knowledge is that it offers ways of avoiding, or greatly reducing, processes of search. The natural enemy of the worker in the field of artificial intelligence is the "combinatorial explosion", and almost his entire craft is concerned with ways of combatting it. The following three examples illustrate the use of stored knowledge to damp off combinatorial explosions.

First, Tables 1 and 2 show the number of combinatorially possible ways in picture-processing of labelling various patterns of intersecting lines, contrasted with the number that are physically possible on the assumption that they arise in retinal projections of three-dimensional scenes composed of plane polyhedral bodies, such as that shown in Fig. 3*a*. The computer program achieves this order of reduction by the use of an appropriate theory. Here I shall review briefly a subset of the theory, adequate for interpreting line drawings of plane-surfaced polyhedra, with trihedral vertices only and without shadows. In this way the flavour can be imparted of the kind of reasoning involved in more complex cases.

Each line in such a drawing can be assigned to one or another of various possible causes: it corresponds to a convex edge, a concave edge, or to an edge formed by two surfaces, only one of which is visible. A corresponding label can be attached to each line, as has been done in Fig. 3*b* using Huffman's conventions[36]. The remarkable fact emerges from Huffman's analysis that only a few of the combinatorially possible ways of labelling such drawings correspond to physically possible structures in the outside world: only twelve distinct configurations of lines around vertices are possible. A computer program can use the theoretical constraints to process the picture, by searching through the space of possible labellings for those which are legal (that is, do not entail that any line should receive two different labels) under the constraints.

Second, Table 3 contrasts the number of topologically possible molecular graphs corresponding to given empirical formulae with the number of candidate interpretations remaining after the Heuristic DENDRAL program has applied its stored theory of chemical stability. The program constructs, using evidence of various kinds, a "GOODLIST" of substructures which must appear in any structure hypothesized by the program and a "BADLIST" of substructures which must not appear. As a simple example, at a given stage down a search tree might be the partial hypothesis —CH_2—O—CH_2— and a possible next move for the structure-generator procedure might be to attach a terminal carbon, forming —CH_2—O—CH_2—CH_3. But unless the data contains peaks at 59 and at M-15 this continuation is forbidden. Again, the structure-generator can be made to handle as a, "super-atom" a fragment indicated by the mass spectrum. Additional opportunities to do this arise when the presence of methyl super-atoms can be inferred from nuclear magnetic resonance data, when available.

Third, McCarthy's problem of the multilated checkerboard[37] is quintessential to the point here discussed. The squares at opposite corners of an 8×8 checkerboard are removed, leaving sixty-two squares. Thirty-one dominoes are available, each of such a size and shape as to cover exactly two adjacent squares of the checkerboard. Can all the sixty-two

Table 1 A Labelling Scheme

+	1 Convex edge
→	2 Obscuring edges—obscuring body lies to right of arrow's direction
←	3 Obscuring edges—obscuring body lies to right of arrow's direction
C→	4 Cracks—obscuring body lies to right of arrow's direction
C←	5 Cracks—obscuring body lies to right of arrow's direction
↑	6 Shadows—arrows point to shadowed region
↓	7 Shadows—arrows point to shadowed region
−	8 Concave edge
=→	9 Separable concave edges—obscuring body lies to right of arrow's direction—double arrow indicates that three bodies meet along the line
=←	10 Separable concave edges—obscuring body lies to right of arrow's direction—double arrow indicates that three bodies meet along the line
=×	11 Separable concave edges—obscuring body lies to right of arrow's direction—double arrow indicates that three bodies meet along the line

Reproduced from ref. 42.

Table 2 Comparison of Number of Combinatorially Possible Labellings with the Number that are Physically Possible

	Approximate number of combinatorially possible labellings	Appoximate number of physically possible labellings.
	2,500	80
	125,000	70
	125,000	500
	125,000	500
	6×10^6	10
	6×10^6	300
	6×10^6	100
	6×10^6	100
	6×10^6	100
	3×10^8	30

Reproduced from ref. 42.

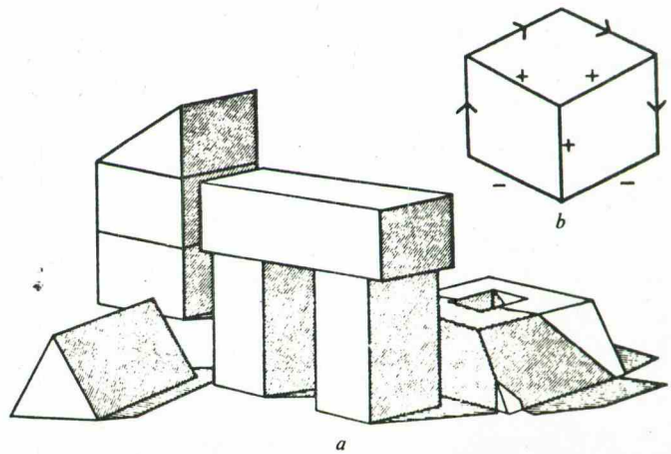

Fig. 3 *a*, A complex three-dimensional scene; *b*, Huffman labels for a cube. Plus implies a convex edge, minus implies concave, and an arrow implies that only one of the edge-forming surfaces is visible.

Table 3 Comparison of the Number of Topologically Possible Molecular Graphs Corresponding to Given Empirical Formulae with the Number of Candidate Interpretations Remaining after the Heuristic DENDRAL Program has Applied its Stored Theory of Chemical Stability

		Number of isomers	Number of inferred isomers A	B
Thiol	1-nonyl	405	89	1
	n-decyl	989	211	1
	n-dodecyl	6,045	1,238	1
Thioether	di-n-pentyl	989	12	1
	di-n-hexyl	6,045	36	1
	di-n-heptyl	38,322	153	1
Alcohol	n-tetradecyl	38,322	7,639	1
	3-tetradecyl	38,322	1,238	1
	n-hexadecyl	151,375	48,865	1
Ether	Di-n-octyl	151,375	780	1
	bis-2-ethylhexyl	151,375	780	21
	di-n-decyl	11,428,365	22,366	1
Amine	n-octadecyl	2,156,010	48,865	1
	N-methyl-n-octyl-n-nonyl	2,156,010	15,978	1
	N,N-dimethyl-n-octadecyl	14,715,813	1,284,792	1

A, Inferred isomers when only mass spectrometry is used; B, inferred isomers when the number of methyl radicals is known from nuclear magnetic resonance data. Based on ref. 20.

squares be exactly covered by some tessellation of the thirty-one dominoes ?

However sophisticated the search procedure which a heuristic program might use to attack this problem by trial and error, the combinatorics of the problem will defeat it. If the reader is unsure of this, let him mentally enlarge the board to, say, 80×80, or $10^8 \times 10^8$. But so long as the dimensions of the board are both of even, or both of odd, length (such boards are called "even" boards) then the problem stays the same for any solver armed with certain crucial pieces of knowledge, namely: that the two squares which are removed from opposite corners of an even board must be of the same colour, and that each domino must cover exactly one white and one black square. The problem now falls apart. The mutilated checkerboard cannot be covered.

To discover formal schemes within which such key facts can automatically be mobilized and their relevance exploited in an immediate and natural fashion is closely bound up with what was earlier referred to as "the representation problem". A familiar example is that certain representations of the game of Nim trivialize the calculation of a winning strategy; but the program capable of inventing such representations is yet to be devised.

Progress Towards an ICS

Two years ago I discussed in *Nature*[27] the possibility of implementing in software an Integrated Cognitive System (ICS). The attainment on a laboratory scale of a "working model", it was suggested, could be used as an indicator of ultimate feasibility. A working model of an ICS, as a minimal set of requirements, should be able: to form an internal representation of its task environment, summarizing the operationally relevant features; to use the representation to form plans of action, to be executed in the task environment; to perform directed perceptual sampling of the environment to switch execution along conditional branches of the plan; to recover from error state when execution fails; to cope with complex and ill-structured environments; to be told new goals and to work out its own approaches to them; and to use the record of past failures and successes to revise and extend the representation inductively.

A computer program which was not able to do most of the above, however excellent a feat of software technology it might be, would not count as an artificial intelligence program. The guidance software for the Apollo on-board computer, written for NASA by Draper Laboratories (J. Moore, privately circulated report, Department of Computational Logic, University of Edinburgh) and charged with the task of getting the spacecraft to the Moon and back, is disqualified on this criterion. On the one hand, it is an acknowledged masterpiece, and on the other, in common with other and lesser automatic control systems, it scores a significant mark only for the third item in the above list.

The on-board computer does not need to plan because hand-coded routine have been provided for all probable situations—analogous, perhaps, to the elaborate, but essentially reflex, nervous system of an insect. The reason for regarding the Apollo on-board system as sub-intelligent is thus concerned with the nature of the internal model which it has of its environment. More than a quarter of a century ago Craik[38] first called attention to the crucial role in thought and perception of internal models. The world of the Apollo computer is so simple and determinate that its behaviour can be completely characterized by computationally simple equations. These equations, which comprise the system's "internal model" in Craik's sense, capture the dynamics of all possible configurations of the objects of its world, and supply all information needed about their interactions and properties.

But consider the mission: not to go to the Moon and back, but the much harder one of going down to the tobacconist and back. By contrast with the space mission, the task environment is exceedingly complex and "messy" and the unexpected lurks at every point of the route (the stairs may be swept, unswept, blocked . . . , the front door may be open, shut, locked . . . , the weather may be bright, dull, wet, windy . . . and so on). Alternatively, and only a little less taxing (at least the environment does not contain other autonomous beings to worry about), consider the mission of a Mars Rover vehicle, such as that already envisaged by NASA[39] and by the space section of the USSR Academy of Sciences (N. Zagoruiko, personal communication). Arising from the fact that it is not possible to pre-program solutions to all problems which might arise while exploring an unknown terrain, a specific ten-year programme of machine intelligence research is regarded as a necessary preliminary condition for putting such operational vehicles into commission. Note that if such a vehicle is to handle all the tasks of autonomous exploration, and assembly and use of instruments, which will be demanded of it, then it must score seven out of seven on the criteria posed earlier.

That achievement lies in the future. How do matters stand today with regard to "working models" ? Each of the seven capabilities listed can now be found in one or another experimental system, and there are some systems which exhibit many, or even most, of them. Unfortunately the most interesting capability of all, central to the phenomenon of intelligence, is the one which is still the least well understood—namely inductive generalization. Yet significant progress has been made[16,40].

In summary, incomplete systems are becoming commonplace and complete "working models", at the most primitive level, now seem not very far off. The likely technological lag before such systems might be upgraded to near-human intellectual performance is a topic for separate consideration.

Implications and Forecasting

It would plainly be desirable to find some objective basis for predicting the rate of development and social impact of machine intelligence. An objective basis is lacking at present and it is only possible to record samples of subjective opinion and to categorize lines of enquiry which more objective studies might follow. Fig. 4 summarizes some of the results of an opinion poll taken last year among sixty-seven British and American computer scientists working in, or close to, the machine intelligence field.

In answer to a question not shown in Fig. 4, most considered that attainment of the goals of machine intelligence would cause human intellectual and cultural processes to be enhanced rather than to atrophy. Of those replying to a question on the risk of ultimate "takeover" of human affairs by

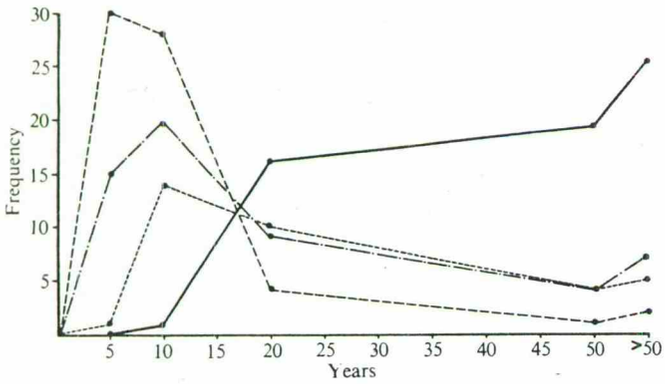

Fig. 4 Opinion poll on machine intelligence. Estimated number of years before: ——, computing system exhibiting intelligence at adult human level; – – –, significant industrial spin-off; — · —, contributions to brain studies; - - -, contributions from brain studies to machine intelligence.

intelligent machines, about half regarded it as "negligible", and most of the remainder as "substantial" with a few voting for "overwhelming".

A working party recently convened under the auspices of the Rockefeller Foundation at Villa Serbelloni, Lake Como, on June 11 to 15, 1972, considered the gradations through which complex information systems might evolve in the future, ranging from contemporary industrial control systems, and "data look-up" retrieval, to autonomous computer networks developed for controlling urban functions (telephones, electricity distribution, sewage, traffic, police, banking, credit systems, insurance, schools, hospitals, and so on). The backbone of such systems will develop anyway, by straightforward elaboration of conventional computing technology, including the integration of the various computational networks into total systems. It seems likely that such systems will also ultimately incorporate autonomous planning and decision-taking capabilities, derived as "spin-off" from developments based on artificial intelligence in, for example, space and oceanographic robotics. A danger could then arise of city dwellers becoming dependent on systems which could no longer be fully understood or controlled. Counter-measures to such dangers might include the introduction of auditing procedures for computer programs, research on program-understanding programs, and system-understanding systems generally, and, finally, the advent of programs to teach the users of intelligent systems.

On the other side of the balance sheet, the working party took preliminary note of several anticipated benefits. The mechanization of industrial production has been associated in the past with the imposition of a deadening uniformity of design. Automated intelligence in the factory could offer the possibility of restoring the diversity and the "one off" capability originally associated with human craftsmanship. Related to this is the introduction of computer aids for the artist, composer, writer, architect and mathematician. Even the ordinary hobbyist might be enabled to perform feats which would today seem daunting or bizarre—building his own house, publishing his own writings, for example. The possible effects on computer-aided education have been stressed by others[41]: advances in this area will be of value not only to the young but also to older people as a means of acquiring new skills.

The formulation of an outline scheme of topics, and the compilation of relevant documents, represents an early stage of a study expected to occupy a number of years. Technical developments which occur in the intervening period will doubtless give such studies a firmer basis.

[1] Lighthill, M. J., *Artificial Intelligence: a general survey* (to be published by the Science Research Council, London).
[2] Turing, A. M., in *Machine Intelligence 5* (edit. by Meltzer, B., and Michie, D.), 3 (Edinburgh University Press, 1969).
[3] Maynard Smith, J., *Evolution*, **6**, 127 (1952), reprinted in *On Evolution* (edit. by Maynard Smith, J.), 29 (Edinburgh University Press, 1972).
[4] Shannon, C. E., *Phil. Mag.*, **41**, 356 (1950).
[5] Turing, A. M., in *Faster than Thought* (edit. by Bowden, B. V.), 288 (Pitman, London, 1953).
[6] Newell, A., Shaw, J. C., and Simon, H. A., *IBM J. Res. Dev.*, **2**, 320 (1958).
[7] Samuel, A. L., *IBM J. Res. Dev.*, **3**, 210 (1959).
[8] Michie, D., Ross, R., and Shannan, G. J., in *Machine Intelligence 7* (edit. by Meltzer, B., and Michie, D.), 141 (Edinburgh University Press, 1972).
[9] Doran, J. E., and Michie, D., *Proc. Roy. Soc.*, A, **294**, 235 (1966).
[10] Hart, P., Nilsson, N. J., and Raphael, B., *IEEE Trans. on Sys. Sci. and Cybernetics*, SSC-4, 100 (1968).
[11] Michie, D., and Ross, R., in *Machine Intelligence 5* (edit. by Meltzer, B., and Michie, D.), 301 (Edinburgh University Press, 1969).
[12] Pohl, I., *Artificial Intelligence*, **1**, 193 (1970).
[13] Michie, D., in *Artificial Intelligence and Heuristic Programming* (edit. by Findler, N. V., and Meltzer, B.), 101 (Edinburgh University Press, 1971).
[14] Fikes, R. E., Hart, P. E., and Nilsson, N. J., *Artificial Intelligence*, **3**, 251 (1972).
[15] Fikes, R. E., and Nilsson, N. J., *Artificial Intelligence*, **2**, 189 (1971).
[16] Winston, P. H., thesis, MIT (1970), reprinted as *MAC-TR-76* (MIT, Project MAC, 1970).
[17] Bobrow, D. G., thesis, MIT (1964), reprinted in *Semantic Information Processing* (edit. by Minsky, M.) (The MIT Press, Cambridge, Mass., 1968).
[18] Woods, W. A., Kaplan, R. M., and Nash-Webber, B., *BBN Report No. 2378* (Bolt, Beranek, and Newman, Cambridge, Mass. 1972).
[19] Winograd, T., thesis, MIT (1970), reprinted in revised form as *MAC-TR-84* (MIT, Project MAC, 1971); also available as *Understanding Natural Language* (Edinburgh University Press, 1972).
[20] Feibenbaum, E. A., Buchanan, B. G., and Lederberg, J., in *Machine Intelligence 6* (edit. by Meltzer, B., and Michie, D.), 165 (Edinburgh University Press, 1971).
[21] McCarthy, J., and Hayes, P. J., in *Machine Intelligence 4* (edit. by Meltzer, B., and Michie, D.), 463 (Edinburgh University Press, 1969).
[22] Ryle, G., *The Concept of Mind* (Hutchinson, London, 1949).
[23] Green, C. C., *Proc. Intern. Joint. Conf. Art. Intell.* (edit. by Walker, D. E., and Norton, L. M.), 219 (Washington DC, 1969).
[24] Raphael, B., in *Artificial Intelligence and Heuristic Programming* (edit. by Findler, N. V., and Meltzer, B.), 159 (Edinburgh University Press, 1971).
[25] Hayes, P. J., in *Machine Intelligence 6* (edit. by Meltzer, B., and Michie, D.), 495 (Edinburgh University Press, 1971).
[26] Burstall, R. M., Collins, J. S., and Popplestone, R. J., *Programming in POP-2* (Edinburgh University Press, 1971).
[27] Michie, D., *Nature*, **228**, 717 (1970).
[28] Hewitt, C., thesis, MIT (1971), reprinted as *Art. Intell. Mem. 258* (MIT, Artificial Intelligence Laboratory, 1972).
[29] Rulifson, J. F., Waldinger, R. J., and Derksen, J. A., *Technical Note 48* (Stanford Research Institute, Artificial Intelligence Group, 1970).
[30] McDermott, D. V., and Sussman, G. J., *Art. Intell. Mem. 259* (MIT, Artificial Intelligence Laboratory, 1972).
[31] Fikes, R. E., Hart, P. E., and Nilsson, N. J., in *Machine Intelligence* (edit. by Meltzer, B., and Michie, D.), 405 (Edinburgh University Press, 1972).
[32] Shirai, Y., and Suwa, M., *Proc. Second Intern. Joint Conf. Art. Intell.*, 80 (British Computer Society, London, 1971).
[33] Will, P. M., and Pennington, K. S., *Artificial Intelligence*, **2**, 319 (1971).
[34] Ejiri, M., Unon, T., Yoda, H., Goto, T., and Takeyasu, K., in *Proc. Second Intern. Joint Conf. on Art. Intell.*, 350 (British Computer Society, London, 1971).
[35] Tan, S. T., *Research Memorandum* MIP-R-98 (University of Edinburgh, School of Artificial Intelligence, 1972).
[36] Huffman, D. A. in *Machine Intelligence 6* (edit. by Meltzer, B., and Michie, D.), 295 (Edinburgh University Press, 1971).
[37] McCarthy, J., *Memo No. 16* (Stanford University, Stanford Artificial Intelligence Project, 1964).
[38] Craik, K. J. W., *The Nature of Explanation* (Cambridge University Press, 1952).
[39] Choate, R., and Jaffe, L. D., in *Proc. First National Conference on Remotely Manned Systems (RMS)* (forthcoming).
[40] Plotkin, G., in *Machine Intelligence 5* (edit. by Meltzer, B., and Michie, D.), 153 (Edinburgh University Press, 1969); also in *Machine Intelligence 6* (edit. by Meltzer, B., and Michie, D.), 101 (Edinburgh University Press, 1971).
[41] Papert, S., *Art. Intell. Mem. 247* (MIT, Artificial Intelligence Laboratory, 1971).
[42] Winston, P. H., in *Machine Intelligence 7* (edit. by Meltzer, B., and Michie, D.), 431 (Edinburgh University Press, 1972).