General AI

Jocelyn Paine*

January 12, 1996

Contents

This is the printed version of these notes. A World-Wide Web version can be found by following the **Students** link on my home page, `http:`/sable.ox.ac.uk/ popxṪhis is almost identical, but contains links to some of the AI people and topics mentioned here.

*`popx@vax.ox.ac.uk`

1 The course

The AI Finals course consists of four topics: General AI, Vision, Language, and Connectionism. You can arrange to do either two or four tutorials in each topic, subject to a maximum of eight tutorials in total. (Actually, most colleges will let you take a few extra tutes if you want to.) This handout contains the notes for the General AI tutes, for those taking only two. They deal mainly with what is known as "classical" or "symbolic" — opposed to connectionist — AI.

1.1 Books

I should have given you a separate handout which tells you where to find AI books and journals. If you can't find a book or photocopy from the Psychology library, you should ask the librarian. Someone may have borrowed it, or you may find that some of the newer accessions haven't yet been catalogued. To reduce strain on the library here, ask your college librarian to buy the books I've cited, and also make a request to the RSL if it doesn't have them. I've included shelfmarks for all the references given in these notes.

1.2 How to find me

If you can't find me around the Department, I may be in the Computing Service or out of Oxford. The best thing to do is to Email me on `popx@vax.ox.ac.uk` . If you don't know how to use Email, ask Dave Popplewell in the Department.

1.3 Finals papers

Even if you are interested in AI for its own sake, you may want to have some goal in mind when reading. Go to the library *now*, make copies of all the AI Finals papers, and check through them. You may not be able to answer the questions yet, but you should review them repeatedly as you go through the tutorials, lectures, and practicals.

You should also practise writing sample answers well before Trinity of your third year. Bear in mind that a lot of the information you'll be given is intended to put the subject in its historical context and to present basic principles. You won't be able to answer Finals questions merely by replaying it verbatim.

Finally, you should supplement these tutorials from lectures or other sources if you want a good chance of being able to answer questions on symbolic AI. The idea is that if you take the complete set of four, you'll get detailed coverage. If you take just the two tutes here, you have probably elected to cover one of the other parts of the syllabus in depth: the main part of these notes will then give you a broad but shallow survey of symbolic AI. I have however added a number of footnotes with suggestions for *optional* further reading, should you want to follow up a topic further. The footnoted references which I think contain the most useful material are starred.

2 Introduction — what is AI? The design stance

The subject covers a lot of ground — an eight-hour AI course is as hopeless an undertaking as an eight-hour psychology course. Many universities allow you to spend your entire three years studying for a degree in the subject. Oxford, of course, does not. Worse, not only is AI a large subject, but it is horribly fragmented. With its lack of general principles, there's a lot of disagreement about aims, methods, and philosophy. Bear this in mind when you compare one person's work with another's.

Approaches to AI differ widely, depending not least on whether the researcher is interested in cognitive science or in engineering. However, I think most would agree with the following definition, given by Aaron Sloman [Computers and Thought 1989]:

a The empirical study of existing intelligent systems. This includes animals as well as people.

b The theoretical study of possible intelligences, and the general principles underlying them.

c Using the above approaches to solve practical problems, i.e.

> **c.1** Trying to deal with the problems of existing intelligences — for example, learning problems and emotional disorders.
>
> **c.2** Designing useful new intelligent (or semi-intelligent) machines.

The definition above is a clear description of AI's objectives, but it's harder to say what AI people should be doing in order to reach them. Different parts of the subject seem to be mutually incompatible, connectionism versus symbolic AI being the best-known example. There are incompatibilities on a smaller scale too, for example the split between classical and non-representational or behaviour-based AI that I'll mention at the end of these notes. It may be that, like architecture, AI can be no more than a collection of tools and techniques. Or it may be that there can be found some unifying concepts which underly the whole of the subject.

As far as psychology is concerned, an important way to look at AI is from the **design stance**. Suppose we set out to design a person from scratch. How would we solve the problems of vision language, and hearing? How could we build a problem solver which balances the need to maintain concentration on existing goals, while being alert to new sensory data but yet not becoming completely distracted by it? What steps are logically necessary in reconstructing object information from images, and how do these relate to the physics of light and the geometry of space? Given that we have limited computational resources, how can we best allocate them between different tasks? In vision, is it necessary to identify objects at all, or can we sometimes make do with simpler processing? Do we need to store beliefs in a logic-based language, or to reason logically? How can we predict how the people around us will behave, without running a complete simulation of their own minds? In other words, when we study AI, we take an engineering approach to the mind.

This is an important meta-idea which you should always bear in mind. However, I haven't had time to reorient these notes around it, and I shall approach AI historically, facing the variety of techniques and methods from the start. So I'll now present a series of papers published between 1950 and 1994, together with a book on the history of AI.

3 Early history

Please start by reading the following papers. After each, I've noted some of the points which I take to be important. Some of the ideas may seem naïve by now, but try to take them seriously — a lot has happened in 45 years! And when comparing attitudes to AI, note the dates carefully. In some of these references, the papers may be accompanied by commentaries which were written ten or twenty years later.

If you look at the bibliography at the end of these notes, it will seem frighteningly long! I am not asking you to read eveything in it — it contains all the *optional* footnoted references as well as the main reading. Unconventionally, I have referenced works by their titles and not their authors, to help you remember the former.

The first paper, by Turing, is [Computing Machinery and Intell]. Note the following:

- p15: Storage media and electronics have improved vastly, and numbers are now stored in binary, but computers still operate in the same way as described here. Turing's "packet" would now be called a "word" (usually 16 or 32 bits, 2 or 4 bytes). If you don't know how a computer works, make sure you understand this.

- pp 17–19: A computer can be programmed to simulate any discrete-state machine. Is the brain such a machine — see page 27.

- pp 21–22: In the past 35 years, numerous attacks on AI have been based on such limitative mathematical results. [1]

- pp 22: The intuitions behind Jefferson's argument seem similar to those behind Searle's famous attack. [2]

- p 31: The tension between learning from experience and learning by being told (or programmed, or fed textbooks).

- pp 33: The *order* in which alternative reasoning steps are investigated is important. We'll come across this again …

I cite the brief reviews in [Mechn of Thought Processes: Report] and [Trends and Discoveries] to give an impression of the variety of approaches already being taken by 1960, including both neurally-inspired and symbolic methods. Note particularly Rosenblatt's Perceptron. This is the forerunner of the back-propagation networks which became so popular in the mid-Eighties and which you'll probably cover in Edmund Rolls' tutorials. [3] [4]

3.1 Connectionism

Because this is covered in the other tutorials, I won't say much about it here. Levels of detail in neural modelling varied then and still do: the neurons of Grey Walter's turtles tried to be more biologically realistic than those of the perceptron. It's worth noting that there are now many varieties of connectionism. The back-prop net is not representative of them all, so don't assume that all connectionist systems share the same properties and limitations. The AI photocopy box contains articles on a variety of systems. [5]

---

[1] Penrose's criticisms on AI in [Emperor's New Mind] are one of the latest.

[2] In [Minds, Brains and Programs].

[3] Why did neural-network research dry up in the early 60's? If you can find a copy, it's worth reading the epilogue in the second edition of Minsky and Papert's *Perceptrons*. Failing that, see Chapter 1 of [Brains, Machines and Mathematics, *].

Another reason for the lack of success may be the unreliability of the early simulations. See page 1390 of [Machines which learn].

[4] Two other popular-science papers [An Imitation of Life, A Machine that Learns, *], from 1950–1951, describe little electric "turtles" that could roam around, seek light and power, and even undergo Pavlovian conditioning. These are an example of a more detailed approach to neural modelling: in this case, the neurons were simulated by valves, capacitors and resistors.

Note the references to cybernetics: in the first paper, these include feedback, homeostasis, and desirable states (goals) — e.g. being near, but not too near, a light. A lot of early AI was inspired by such ideas. Page 44 contains what may be the first example of AI hype.

The second paper [A Machine that Learns, *], raises the question of how a learning system should detect significant correlations — and discard what it's learnt if they cease to be significant.

[5] One problem about which there's still much debate is the exact relationship between AI programs and cognitive models which represent their knowledge as strings of disrete symbols (like those I'll cover

3.2 Logic-based AI

As an exemplar of such programs, I want you to look at McCarthy's advice-taking program in more detail. Please read [Programs with Common Sense]. You should find it fairly easy to read. This is partly because it occurs very early in AI history, but also because it remains uncluttered with implementation details, being a proposal for a program that was never built (something not entirely untypical of later AI work). Despite this, the paper is worth taking seriously: McCarthy is one of the founders of AI, and went on to do a lot of work on AI and logic, as well as inventing the Lisp programming language. As with the papers above, you won't be expected to reproduce the workings of this program for Finals. What I want you to note is the general approach implied, that of representing the world as logical propositions.

A problem still largely unsolved is how such deductive systems should distinguish just those premises that it needs from all the irrelevant ones. Some critics still assert that this is impossible in principle, and hence so is this kind of AI: see p 86. [6]

4 Early cognitive theories, search, heuristics, and the combinatorial explosion

4.1 The Logic Theorist

This was one of the first ever implemented AI programs. It's interesting because it was designed in part by Allen Newell, a leading exponent of the symbol-processing approach to cognitive modelling. His work culminated in 1990 with the publication of a book on unified theories of cognition. The paper cited below is one of the first statements of his views on AI as a theory of cognition. So please read [Elements of a Theory...]. Pay most attention to Newell's views of what such a theory should do (pp 151–153) and of its relation to other schools of psychology (pp 163–166).

How does the program actually work? The important point is that you start with a set of axioms of logic from which a theorem is to be proven. These axioms can be thought of as "sentences" describing what you originally know. A theorem is also a sentence, which has to be derived from the axioms.

As well as the axioms, you have a number of *rules of inference* which convert old sentences into new ones. There is a very close analogy to a one-person board game. The axioms are the starting positions; rules of inference are the permitted moves; proving a theorem is equivalent to asking whether a given state of the board is reachable by a sequence of legal moves from one of the starting positions.

In principle, one could *exhaustively generate* all possible sentences from the axioms, stopping when one of them was the same as the theorem to be proven. This is what the authors refer to as the "British Museum algorithm". Because this algorithm generates all possible sentences, it's guaranteed to find a sentence that matches the theorem to be proved, should one exist.

In practice, this is impossible on any present or future computer, because of the *combinatorial explosion* in the number of possible moves. So one needs *heuristics*: methods which save time by looking only at plausible moves. In the jargon of AI, they *prune the search tree*. However, unless the heuristics are to be as slow as the

---

in these notes), and connectionist systems where the knowledge is implicit in synaptic weights and neural activations. See [Artificial Intelligence: Pratt] for a clear two-page statement of one view and some further references.

[6] Dreyfus is one of the best-known critics of the logical approach. See [Mind over Machine]. Daniel Dennett has argued that while the problem of selecting such premises is indeed very hard, it does not signal the death of AI. See [Cognitive Wheels], in which he starts with an elegant little fable of three robots. For one example of a commercially valuable logic-based system designed to advise motorists about accidents, see [POETIC: A system..., *]. This is a good examplar of logic-based symbolic AI.

original exhaustive computation, they can only make a guess at plausibility, relying on hopefully informative but sometimes deceptive surface features. Hence they do not *guarantee* a solution. [7] [8]

5 Planning

I shan't cover this in detail, so this section just gives a brief summary. It sets one reading, again on Newell's work. However, there are numerous footnotes in case you ever want to follow these topics further.

It's worth knowing that the word "planning" usually refers to the following type of problem. You start with the world in a given (known) state. You are able to perform various elementary actions, each of which changes the world in a known way. Your task is to generate some sequence of actions or *plan* which will get the world into some goal state. It's generally assumed that the start state, the goal state, and the effect of each action can be described in logic.

One example is chemical synthesis. Your goal is a product or set of products. Your start state is a set of reagents. Your actions are chemical reactions. A one-person game, or the missionary-and-cannibals problem [9] is another example.

The simplest way to tackle planning is by search. However, even with heuristics, that's very inefficient: they don't provide enough information about your progress towards a goal. An alternative method is *problem reduction*. You break your original goal into a set of simpler *subgoals*. If a subgoal can be achieved by one of the elementary actions, then you've solved that part of the problem. Otherwise, you break it up into yet simpler subgoals ...and so on until you've solved them all. In this formulation, it's necessary to have some knowledge of how likely each elementary action is to take you towards a goal.

The first planning program is generally taken to be the General Problem Solver, usually known just as *GPS*. It was a descendant of the Logic Theorist, and also designed in part by Newell, both as a cognitive model and as an exercise in problem-solving techniques. For an account of GPS, please read *GPS, a program that simulates human thought*, in [Computers and Thought 1963]. Concentrate on Newell's motivations (note the conclusion on p 293); if you have problems understanding the program description, try the footnote references. [10]

Another planner, written just after GPS and using essentially the same techniques, was Strips. After these came many more, with names like Abstrips, Noah, Nonlin,

---

[7] You may find it easier to follow the account in [The Logic Theory Machine]. Also, it may help to draw some tree diagrams. If you want further explanation of search and heuristics, then there's a very good account, with diagrams and examples, in [Learning and Problem Solving, *], pp 8–24.

[8] One subject where it's very important to reduce search is chess. As well as all the work on faster chess machines, there's been a lot of interest in cognitive science on the psychology of chess players. See *Human chess skill* by Neil Charness in [Chess skill in man and machine], and the papers he cites, particularly the Chase and Simon reference.

Newell has worked on chess — he makes some suggestions for a logic-based chess-program in his 1957 article *The Chess Machine* in [**?**]. Pages 73–76 are a very clear statement of the search problems involved. In *Chess-playing programs and the problem of complexity*, from [Computers and Thought 1963], Newell, Shaw and Simon compare the performance of the then existing chess programs. This article has a good explanation of minimaxing, also covered in [Learning and Problem Solving, *].

[9] [Learning and Problem Solving, *]

[10] Most AI textbooks contain descriptions of GPS — if you aim to write about any of the planning work, you should certainly include it. Boden gives a clear non-technical account in pages 354–357 of [AI and Natural Man]. Winston's account in [Artificial Intelligence: Winston, *] is probably one of the best.

See page 212 of [Machines who Think] for some informal comments by Newell and Simon on the spark of intuition that produced GPS. Until Crevier's book [AI: The Tumultuous..., *] appeared, this was the only general history of AI ever written. It contains a number of interviews with Newell, Minsky, and other founders of AI.

and so on. [11]

It gradually came to be realised that such programs did not perform terribly well when faced with the real world of robotics, partly because of imprecision in sensing and moving, but not least because even when everything is known exactly, the methods are still horribly inefficient. Later work in planning and problem-solving has moved away from these a-priori logic-based methods, and has gone in other directions. One of these, *case-based planning*, entails storing details of existing solutions in memory, and trying to adapt them. [12]

Some people have been even more radical. The classical planning researchers all assumed that their programs would, via some kind of perceptual system, build up a symbolic model of the world in memory. From this *world model*, together with symbolic representations of its goals, it would derive a plan which would then by executed by a motor system. Some researchers have repudiated entirely the classical assumption that the program should build faithful symbolic descriptions of its environment. I'll set some readings on this at the end of these notes.

6 Production systems

Production systems are collections of IF-THEN rules. In the simplest case, the rules may be straight S-R rules, mapping directly to some motor action. Usually though, they operate on *working memory*, adding or deleting clusters of symbols which can then be tested for by other rules. So the rules are usually of the form

```
IF you have this pattern of symbols in STM
THEN add or delete these new symbols
```

or

```
IF you have this pattern of symbols in STM
THEN carry out this action
```

Essentially, production systems generalise the notion of stimulus-response system: the response may be not only an action on the outside world, but also a change to mental state.

Technologically, production systems are not very interesting. Most just follow a simple "forward-chaining" procedure which enables them to reason forwards from premises to conclusions. Their interest lies in their use as psychological models and once again, Newell has been a major exponent. In moving from GPS to production systems, he has focussed on the issue of *control*. In computer science, this refers to the way that a program jumps from one statement to another — its "attention strategy". Whereas GPS' control strategy was built in to it, the control in a production system is driven much more by the problem and the data we have. So we focus much more than with GPS on the information that's present in memory, and how it guides us in switching from one problem-solving operation to another.

---

[11] How did planning evolve from here? You can get a quick survey by reading the chapter introductions in [Readings in Planning]: the Foreword, then pages 57, 109–110, 187–188, 289–290, 391–392, 521, 579, 647–649.

It was realised early on that one of the key problems was interactions between subgoals. For example, if your goal is to make the table look nice and set it for tea, then polishing it will disturb the goal of setting it, since you'll have to wait for the polish to dry. The first program to attempt interacting goals was Hacker. It, essentially, planned as though the goals were independent, and then tried to correct faults in the resulting plans. See pages 286–297 and 360–366 of [AI and Natural Man].

[12] People from conventional AI often underestimate the material problems a robot or animal would face in the real world. These include motion control, perception, noise, and dealing with the opposition. Charniak and McDermott pages 527–543 describe some of the problems faced by (a) assembly robots and (b) games players.

To start, find out how they work. For this, the account on pages 190–199 of volume 1 of [Handbook of AI] is good. They're also described in the final part of [Learning and Problem Solving] and in most AI textbooks.

Now to some cognitive models. Production systems have been used to models childrens' development, and I'll ask you to read on this next. Find a copy of [Computers and Thought 1989]. The start of Chapter 7 places production systems in a psychological context, while pages 226–231 in Chapter 8 describe their use in modelling childrens' subtraction. Follow up with the Young and O'Shea article in *Cognitive Science* for 1981 that they reference. Note: there is a misprint in the book that sets this article in 1982.

Another view is given in Richard Young's chapter on *Production Systems for Modelling Human Cognition* in [Expert systems in microelectronic age]. Pay particular attention to the three levels of modelling he distinguishes, on page 43. Young's work on seriation is described at the end of [Learning and Problem Solving]. You should try to hand-simulate at least a few cycles of one of his programs. [13] [14]

7 Interlude

This is the end of the first week's tutorial. Please select *one* of the following questions and write an essay on it. On the evidence of past Finals papers, it's probably worth preparing in some depth for a production-system question.

a Discuss the advantages and disadvantages of rule-based models of problem-solving over connectionist models.

b What, if anything, have production systems contributed to developmental psychology?

c Production systems can simulate cognitive tasks. Does it therefore follow that the mind is a production system?

8 Evans, Marr, and the importance of representations

In this section, I'll skip back to 1968 and then run quickly through the 70's, leading up to some work that emphasises the importance of *representation*.

Start with Minsky's introduction in Chapter 1 of [Semantic Information Processing]. This is a typical account of AI from the viewpoint of 1968. Read the chapter once, noting Minsky's views on the important developments within AI, the direction it has taken since 1960, and its relation to psychology.

8.1 Evans' Analogy Program

Now return to pp 16–17 for comments on the Evans Analogy Program, and then read pp 271–288 in the introduction to Evans' article in the same book. As p 281 makes clear, Evans wrote his program as an experiment in the use of "good" "high-level" representations of simple pictures. This was then an idea still new to computing and AI — now it's old hat. So the Analogy program is worth knowing because it

[13] How did Newell believe production systems could benefit psychology? Read [You can't play 20 questions..., *], and follow it with pages 154–168 and 210–213 from [Computer Models of Mind].

[14] If you intend to write a Finals question on production systems, you should certainly look at SOAR in [Unified Theories of Human Cognition, *] at some stage of the course. Try to read the first four chapters. You will find a short survey of the book in *Behavioural and Brain Sciences* volume 15, number 3, September 1992. This also contains a number of criticisms of SOAR and the idea of unified theories by cognitive psychologists, computer scientists and others.

demonstrates the importance of representation, and why some representations are better than others. By trying to simulate its behaviour, you can get a feel for this.

Evans goes on to describe in detail how his program operates. Apart from the summary on pp 288–290, you'll probably find this account too technical. However, there's an easier one written by Patrick Winston, formerly one of Evan's students at MIT. This is in pp 24–33 of [Artificial Intelligence: Winston]: it includes a discussion of what makes a representation good. So read these pages, and work some examples. How does the representation restrict the class of analogies the program can solve?

One extra point: you should be aware that Winston's account is different from Evans' own, being simplified to make it easier to understand. The program was actually more complex (to make it more efficient). However, the principle, of finding the difference between A and B, and then looking for that figure D_n which differs from C in the same way as B differs from A, is the same.

8.2 Marr

Now please read the introduction and Chapter 1 of [Vision]. Make sure you understand Marr's notion of representation, the difference between his three levels of explanation, and the reasons for his moving to representation-based AI. Marr also writes these ideas up in *Visual Information Processing* from [Psychology of Vision]. You may find this easier to read.

Marr's model illustrates the advantages of having more than one representation, and how different representations make different kinds of information explicit. But his model is only a hypothesis about the brain's workings. The point can be made more forcefully by looking at a man-made construct (a computer program) whose author has chosen the representations and documented the reasons for each, and this is why I started with Evans' program. Note incidentally that Marr, Winston and Evans were all at MIT (though not perhaps all at the same time). [15]

Since you will be doing tutorials on vision, I'll leave the work of Marr and his successors here. But representation is not only important to vision. For a general view of its rôle in AI, see Roger Schank's article in [Foundations of AI]. Schank's work has been particularly concerned with the representation of memory, including the case-based reasoning that I mentioned in connection with planning. I don't know when the paper was written, but it would be typical of his attitude from the late 70's onward. [16]

9 Expert systems

There is no exact definition of an expert system, but it's generally agreed that expert systems:

- Represent their knowledge symbolically. In practice, most systems used IF-THEN rules.
- After giving an answer, can justify or explain it by showing which knowledge they used to come to that conclusion.

[15] For a technological example of an Evans-style relational description, see [Learning shape descriptions] and [The Mechanic's Mate]. The Mechanics Mate was a program designed to reason about (for example) how best to remove nails from wood. In order to do this, it had to look at the work and decode the resulting image into a symbolic description (giving, e.g. the type of nail) which could then be reasoned about when selecting a suitable tool.

[16] John Hallam, in [Computational Theories...], applies Marr's explanatory framework. You may find this a useful example: I haven't checked it.

- In a similar way, can explain why they have asked their user a particular question.
- Can imitate the performance of a human expert in a narrow domain of expertise, such as mortgage tax advice, interpreting mass spectra, or planning computer systems.

Technically speaking, most expert systems are and were (like production systems) relatively uninteresting, being logic-based programs which used simple inference methods to reason over their rules. However, they played a large rôle in making AI visible to business and government. This had an enormous, but not necessarily beneficial, effect on the funding of AI research.

Expert systems also made the study of human-computer interaction more important. A simple rule-based expert system can easily be programmed to generate automatic explanations of how it has reached a conclusion, or why it is asking a question. However, these explanations leave a lot to be desired, being too detailed and low-level. These defects stimulated work on systems which give a more structured overview of their reasoning, and even ones which tried to adapt to their user's level of skill.

The rules that make up an expert system's knowledge are often gleaned from human experts. This spawned a new subject of *knowledge engineering* or *knowledge acquisition*. Psychological skill was important to make sure that the experts told you what they knew, rather than what they thought they knew.

In general, expert systems are not intended as cognitive models. It helps (in some cases), if a system reasons in a way that a human can understand. However, we're not usually aiming to produce a detailed model of the expert, merely a tool which can solve the same problems. Because of this, and because of the technical simplicity of most expert systems, I shan't say much about them. In any case, if you understand production systems, you won't find rule-based expert systems difficult.

For a view from the heights of commercial interest, skim [Guide to Expert Systems]. This was written in 1985, near the top of the expert systems boom. Indeed, the book was a result of this. A new expert systems journal had just started up, intended for a business audience. The book was to be given away (or sold cheaply) to new subscribers with no knowledge of AI, as a way of easing them into the subject. The book contains numerous examples of expert systems and a brief description of inference, and exemplifies the optimistic attitude of its time.

10 Non-representational AI

So what does the typical symbolic AI program or cognitive model look like? Read the forward to [Computers and Thought 1989]. This short piece by Aaron Sloman has been around since at least 1983, as one of the HELP files in the Poplog programming system we use.

Inspired (or not) by the failure of robotics, researchers have started questioning such models. So please read the Preface and Chapter 1 of [Intelligence as adaptive behaviour]. [17]

[17] If you want to add to this, I recommend [Intelligence without Representation], and [An emerging paradigm in robot behaviour, *]. You'll find that the second is useful in giving a concise summary of the classical approach to AI and robotics.

11 Finally

If you want to see how all the above sits inside the rest of AI, we have two popular science books which are a bit oversimplified, but not bad as surveys. I'd recommend trying these sometime during the course. [18]

Then write *one* of the following essay questions. As with production systems, questions on representation have appeared frequently.

a Describe, using Evans' Analogy Program as an example, why information-processing systems (natural and artificial) need multiple representations. How could you extend the program to solve 3-d analogy problems?

b How (if at all) does the notion of representation apply to connectionist systems?

References

[A Machine that Learns] *A Machine that Learns*, W Grey Walter, from *Scientific American*, volume 185, number 2, August 1951. In the RSL; also as PSY AI photocopy W89.

[AI and Natural Man] *AI and Natural Man*, Margaret Boden, Harvester 1977, PSY KD:B 063.

[AI: The Tumultuous...] *AI: the tumultuous history of the search for artificial intelligence*, Daniel Crevier, Basic Books 1993, PSY KH:C 068.

[An Imitation of Life] *An Imitation of Life*, W Grey Walter, from *Scientific American*, volume 182, number 5, May 1950. In the RSL; also as PSY AI photocopy W90.

[An emerging paradigm in robot behaviour] *An emerging paradigm in robot behaviour*, C A Malcolm, Tim Smithers, and John Hallam. DAI research paper 447, 1989. PSY AI photocopy M149.

[Artificial Intelligence: Pratt] *Artificial Intelligence*, Ian Pratt, 1994 ???. PSY ??? just bought.

[Artificial Intelligence: Winston] *Artificial Intelligence (2nd edition)*, P H Winston, 1984, Addison-Wesley. PSY KH:W 72; RSL Comp Bd 18.

[Artificial Life] *Artificial Life*, S Levy, Penguin 1992., PSY KH:L 057.

[18] There's one on the general history of AI: [AI: The Tumultuous..., *]. You can skip anything before Boole and Babbage, as well as speculations about the future of AI: concentrate on the research and commercial achievements Crevier describes, and the flow of ideas. The book is oversimplified, but not bad as a survey. Then contrast with the new approaches described in the chapter on *Real Artificial Life* in [Artificial Life, *]. The rest of this book is also interesting, and I find that it conveys very well the enthusiasm and excitement of a new field.

[Brains, Machines and Mathematics] *Brains, Machines and Mathematics (2nd edition)*, Michael Arbib, Springer-Verlag 1987. RSL Comp Bd 98; Chapter 1 copied as AI box photocopy A52.

[Chess skill in man and machine] *chess skill in man and machine (2nd edition)*, P W Frey (ed), Springer-Verlag 1984. PSY KH:F 089.

[Cognitive Wheels] *Cognitive Wheels*, Daniel Dennett, from *The Robot's Dilemma*, Pylyshyn (ed), Ablex 1987. PSY AI photocopy D74. Also appears in C Hookaway, *Minds, Machines and Evolution*, CUP 1984.

[Computational Theories...] *Computational theories for research in vision and cognition*, John Hallam, Edinburgh University 1987. PSY AI photocopy H157.

[Computer Models of Mind] *Computer Models of Mind*, Margaret Boden, CUP 1988. PSY KD:B 063.

[Computers and Thought 1963] *Computers and Thought*, E Feigenbaum and J Feldman (eds), McGraw-Hill 1963. PSY KH:F 032.

[Computers and Thought 1989] *Computers and Thought*, M Sharples, D Hogg, C Hutchison, S Torrance, and D Young, MIT 1989. PSY KH:S531; RSL M92.E00068; Hooke.

[Computing Machinery and Intell] *Computing Machinery and Intelligence*, Alan Turing, from [Computers and Thought 1963].

[Elements of a Theory...] *Elements of a Theory of Human Problem-solving* Allen Newell and H A Simon, from Psychological Reviews, volume 65, pp 151–166, 1958. PSY journal.

[Emperor's New Mind] *The Emperor's New Mind*, Roger Penrose, OUP 1989. PSY KH:P 38.

[Expert systems in microelectronic age] *Expert systems in the micro-electronic age*, Donald Michie (ed), EUP 1980, PSY KH:M 58.

[Foundations of AI] *The Foundations of AI*, Partridge and Wilks (eds), CUP 1990. PSY KH:P 025.

[Guide to Expert Systems] *The Guide to Expert Systems*, Alex Goodall, Learned Information 1985. RSL Comp BD 36; PSY KH:G 061.

[Handbook of AI] *Handbook of AI*, Barr and Feigenbaum (eds), Kaufmann 1981, PSY KH:B 27.

[Intelligence as adaptive behaviour] *Intelligence as adaptive behaviour*, R D Beer, Academic Press 1990., PSY KH:B 039.

[Intelligence without Representation] *Intelligence without Representation*, R A Brookes, from *Artificial Intelligence*, volume 47, 1991, pp 139–159. PSY; RSL.

[Learning and Problem Solving] *Learning and Problem Solving part 3*, Open University Course D303, Block 4, Units 26–28, Open University 1978. PSY oversize AA:O 2–O M.

[Learning shape descriptions] *Learning Shape Descriptions*, J H Connell and Mike Brady, PSY AI photocopy C117.

[Machines which learn] *Machines which learn*, A M Andrew, from *New Scientist*, volume 4, 27 November 1958. In the RSL; also as PSY AI photocopy A53. This copy is imperfect, since the volume was too thick to open properly near the spine.

[Machines who Think] *Machines who Think*, Pamela McCorduck, Freeman 1979. PSY KH:M 13.

[Mechn of Thought Processes] *Mechanization of Thought Processes*, HMSO 1959. PSY AI KC:N 21, volume 1..

[Mechn of Thought Processes: Report] *Mechanization of Thought Processes: Report*, from *Nature*, volume 188, 24 January 1959. In the RSL; also as PSY AI photocopy D78. This copy is slightly imperfect.

[Mind over Machine] *Mind over Machine*, Dreyfus and Dreyfus, Blackwell 1986. PSY KH:D 81.

[Mind's I] *The Mind's I*, D Hofstadter and D Dennett (eds), Harvester 1981. PSY AA:H 067; Hooke.

[Minds, Brains and Programs] *Minds, Brains and Programs*, John Searle, from *Behavioural and Brain Science*, volume 3, 1980. PSY journals. This paper also appears in [Mind's I]. That version doesn't contain the responses by Searle's critics you'll find here, but it does have a nice commentary by Hofstadter and Dennett which is worth reading.

[POETIC: A system...] *POETIC: A system for gathering and disseminating traffic information*, R Gaizauskas, R Evans, L J Cahill, J Richardson, J Walker, from *Conf. Preprints Int. Conf. AI Applics. in Transportation Engineering*, S G Ritchie and C Hendrikson (eds), June 1992, California. PSY AI photocopy G87.

[Programs with Common Sense] *Programs with Common Sense*, John McCarthy, From [Mechn of Thought Processes], pp 77–91.

[Psychology of Vision] *Psychology of Vision*, C Longuet-Higgins and S Sutherland (eds), Royal Society 1980, PSY oversize BC:R 81. This paper is reprinted from *Phil Trans Royal Society London*, volume B 290, pages 199-218.

[Readings in Planning] *Readings in Planning*, J Allen, J Hendler, A Tate (eds), Morgan Kaufmann 1990. PSY KH:A 427.

[Semantic Information Processing] *Semantic Information Processing*, Marvin Minsky (ed), MIT 1968. PSY BK:M 66.

[The Logic Theory Machine] *Empirical Explorations with The Logic Theory Machine: A Case Study in Heuristics*, Allen Newell, J C Shaw, and H A Simon, from [Computers and Thought 1963].

[The Mechanic's Mate] *The Mechanic's Mate* Mike Brady and P Agre, from *ECAI 84*, T O'Shea (ed), Elsevier 1984. PSY AI photocopy B157.

[Trends and Discoveries] *Trends and Discoveries*, from *New Scientist*, volume 4, 27 November 1958. In the RSL; also as PSY AI photocopy N27. This copy is slightly imperfect.

[Unified Theories of Human Cognition] *Unified Theories of Human Cognition*, Allen Newell, Harvard 1990. PSY BH:N 044.

[Vision] *Vision*, David Marr, Freeman 1982. PSY CH:M 34; Hooke.

[You can't play 20 questions...] *You can't play 20 questions with nature and win*, Allen Newell, from *Visual Information Processing*, W Chase (ed), Academic Press 1973. PSY BC:C 39.