The Computer Revolution In Philosophy (1978)
Aaron Sloman

This chapter is also available in PDF format here.

CHAPTER 1

INTRODUCTION AND OVERVIEW

1.1. Computers as toys to stretch our minds

Developments in science and technology are responsible for some of the best and some of the worst features of our lives. The computer is no exception. There are plenty of reasons for being pessimistic about its effects in the short run, in a society where the lust for power, profit, status and material possessions are dominant motives, and where those with knowledge -- for instance scientists, doctors and programmers -- can so easily manipulate and mislead those without.

Nevertheless I am convinced that the ill effects of computers can eventually be outweighed by their benefits. I am not thinking of the obvious benefits, like liberation from drudgery and the development of new kinds of information services. Rather, I have in mind the role of the computer, and the processes which run on it, as a new medium of self-expression, perhaps comparable in importance to the invention of writing.

Think of it like this. From early childhood onwards we all need to play with toys, be they bricks, dolls, construction kits, paint and brushes, words, nursery rhymes, stories, pencil and paper, mathematical problems, crossword puzzles, games like chess, musical instruments, theatres, scientific laboratories, scientific theories, or other people. We need to interact with all these playthings and playmates in order to develop our understanding of ourselves and our environment that is, in order to develop our concepts, our thinking strategies, our means of expression and even our tastes, desires and aims in life. The fruitfulness of such play depends in part on how complex the toy and the processes it generates, and how rich the interaction between player and toy are.

A modern digital computer is perhaps the most complex toy ever created by man. It can also be as richly interactive as a musical instrument. And it is certainly the most flexible: the very same computer may simultaneously be helping an eight year old child to generate pictures on a screen and helping a professional programmer to understand the unexpected behaviour of a very complex program he has designed. Meanwhile other users may be attempting to create electronic music, designing a program to

translate English into French, testing a program which analyses and describes pictures, or simply treating the computer as an interactive diary. A few old-fashioned scientists may even be doing some numerical computations.

Unlike pet animals and other people (also rich, flexible and interactive), computers are toys designed by people. So people can understand how they work. Moreover the designs of the programs which run on them can be and are being extended by people, and this can go on indefinitely. As we extend these designs, our ability to think and talk about complex structures and processes is extended. We develop new concepts, new languages, new ways of thinking. So we acquire powerful new tools with which to try to understand other complex systems which we have not designed, including systems which have so far largely resisted our attempts at comprehension: for instance human minds and social systems. Despite the existence of university departments of psychology, sociology, education, politics, anthropology, economics and international relations, it is clear that understanding of these domains is currently at a pathetically inadequate level: current theories don't yet provide a basis for designing satisfactory educational procedures, psychological therapies, or government policies.

But apart from the professionals, ordinary people need concepts, symbolisms, metaphors and models to help them understand the world, and in particular to help them understand themselves and other people. At present much of our informal thinking about people uses unsatisfactory mechanistic models and metaphors, which we are often not even aware of using. For instance even people who strongly oppose the application of computing metaphors to mental processes, on the grounds that computers are mere mechanisms, often unthinkingly use much cruder mechanistic metaphors, for instance 'He needed to let off steam', 'I was pulled in two directions at once, but the desire to help my family was stronger', 'His thinking is stuck in a rut', 'The atmosphere in the room was highly charged'. Opponents of the spread of computational metaphors are in effect unwittingly condemning people to go on living with hydraulic, clock-work, and electrical metaphors derived from previous advances in science and technology.

To summarise so far: it can be argued that computers, or, to be more precise, combinations of computers and programs, constitute profoundly important new toys which can give us new means of expression and communication and help us create an ever-increasing new stock of concepts and metaphors for thinking about all sorts of complex systems, including ourselves.

I believe that not only psychology and social sciences but also biology and even chemistry and physics can be transformed by attempting to view complex processes as computational processes, including rich information flow between sub-processes and the construction and manipulating of symbolic structures within processes. This should supersede older paradigms, such as the paradigm which represents processes in terms of equations or correlations between numerical variables.

This paradigm worked well for a while in physics but now seems to dominate, and perhaps to strangle, other disciplines for which it is irrelevant. Apart from computing science, linguistics and logic seem to be the only sciences which have sharply and successfully broken away from the paradigm of 'variables, equations and correlations'. But perhaps it is significant that the last two pretend not to be concerned with processes, only with structures. This is a serious limitation, as I shall try to show in later chapters.

1.2. The Revolution in Philosophy
Well, suppose it is true that developments in computing can lead to major advances in the scientific study of man and society: what have these scientific advances to do with philosophy?

The very question presupposes a view of philosophy as something separate from science, a view which I shall attempt to challenge and undermine later, since it is based both on a misconception of the aims and methods of science and on the arrogant assumption by many philosophers that they are the privileged guardians of a method of discovering important non-empirical truths.

But there is a more direct answer to the question, which is that very many of the problems and concepts discussed by philosophers over the centuries have been concerned with *processes,* whereas philosophers, like everybody else, have been crippled in their thinking about processes by too limited a collection of concepts and formalisms. Here are some age-old philosophical problems explicitly or implicitly concerned with processes. How can sensory experience provide a rational basis for beliefs about physical objects? How can concepts be acquired through experience, and what other methods of concept formation are there? Are there rational procedures for generating theories or hypotheses? What is the relation between mind and body? How can non-empirical knowledge, such as logical or mathematical knowledge, be acquired? How can the utterance of a sentence relate to the world in such a way as to say something true or false? How can a one-dimensional string of words be understood as describing a three-dimensional or multi-dimensional portion of the world? What forms of rational inference are there? How can motives generate decisions, intentions and actions? How do non-verbal representations work? Are there rational procedures for resolving social conflicts?

There are many more problems in all branches of philosophy concerned with processes, such as perceiving, inferring, remembering, recognising, understanding, learning, proving, explaining, communicating, referring, describing, interpreting, imagining, creating, deliberating, choosing, acting, testing, verifying, and so on. Philosophers, like most scientists, have an inadequate set of tools for theorising about such matters, being restricted to something like common sense plus the concepts of logic and physics. A few have clutched at more recent technical developments, such as concepts from control theory (e.g. feedback) and the mathematical theory of games (e.g. payoff matrix), but these are hopelessly deficient for the tasks of philosophy, just as they are for the task of psychology.

The new discipline of artificial intelligence explores ways of enabling computers to do things which previously could be done only by people and the higher mammals (like seeing things, solving problems, making and testing plans, forming hypotheses, proving theorems, and understanding English). It is rapidly extending our ability to think about processes of the kinds which are of interest to philosophy. So it is important for philosophers to investigate whether these new ideas can be used to clarify and perhaps helpfully reformulate old philosophical problems, re-evaluate old philosophical theories, and, above all, to construct important new answers to old questions. As in any healthy discipline, this is bound to generate a host of new problems, and maybe some of them can be solved too.

I am prepared to go so far as to say that within a few years, if there remain any philosophers who are not familiar with some of the main developments in artificial intelligence, it will be fair to accuse them of professional incompetence, and that to teach courses in philosophy of mind, epistemology, aesthetics, philosophy of science, philosophy of language, ethics, metaphysics, and other main areas of philosophy, without discussing the relevant aspects of artificial intelligence will be as irresponsible as giving a degree course in physics which includes no quantum theory. Later in this book I shall elucidate some of the connections. Chapter 4, for example, will show how concepts and techniques of philosophy are relevant to AI and cognitive science.

Philosophy can make progress, despite appearances. Perhaps in future the major advances will be made by people who do not call themselves philosophers.

After that build-up you might expect a report on some of the major achievements in artificial intelligence to follow. But that is not the purpose of this book: an excellent survey can be found in Margaret Boden's book. *Artificial Intelligence and Natural Man,* and other works mentioned in the bibliography will take the interested reader into the depths of particular problem areas. (Textbooks on AI will be especially useful for readers wishing to get involved in *doing* artificial intelligence.)

My main aim in this book is to re-interpret some age-old philosophical problems, in the light of developments in computing. These developments are also relevant to current issues in psychology and education. Most of the topics are closely related to frontier research in artificial intelligence, including my own research into giving a computer visual experiences, and analysing motivational and emotional processes in computational terms.

Some of the philosophical topics in Part One of the book are included not only because I think I have learnt important things by relating them to computational ideas, but also because I think misconceptions about them are among the obstacles preventing philosophers from accepting the relevance of computing. Similar misconceptions may confuse workers in AI and cognitive science about the nature of their discipline.

For instance, the chapters on the aims of science and the relations between science and philosophy attempt to undermine the wide-spread assumption that philosophers are doing something so different from scientists that they need not bother with scientific developments and *vice versa*. Those chapters are also based on the idea that developments in science and philosophy form a computational process not unlike the one we call human learning.

The remaining chapters, in Part Two, contain attempts to use computational ideas in discussing some problems in metaphysics, philosophy of mind, epistemology, philosophy of language and philosophy of mathematics. I believe that further analysis of the nature of number concepts and arithmetical knowledge in terms of symbol-manipulating processes could lead to profound developments in primary school teaching, as well as solving old problems in philosophy of mathematics.

In the remainder of this chapter I shall attempt to present, in bold outline, some of the main themes of the computer revolution, followed by a brief definition of ''Artificial Intelligence''. This will help to set the stage for what follows. Some of the themes will be developed in detail in later chapters. Others will simply have to be taken for granted as far as this book is concerned. Margaret Boden's book and more recent textbooks on AI fill most of the gaps.

1.3. Themes from the Computer Revolution
1. Computers are commonly viewed as elaborate numerical calculators or at best as devices for blindly storing and retrieving information or blindly following sequences of instructions programmed into them. However, they can be more accurately viewed as an extension of human means of expression and communication, comparable in importance to the invention of writing. Programs running on a computer provide us with a medium for thinking new thoughts, trying them out, and gradually extending, deepening and clarifying them. This is because, when suitably programmed, computers are devices for constructing, manipulating, analysing, interpreting and transforming symbolic structures of all kinds, including their own programs.

2. Concepts of 'cause', law', and 'mechanism', discussed by philosophers, and used by scientists, are seriously impoverished by comparison with the newly emerging concepts.

The old concepts suffice for relatively simple physical mechanisms, like clocks, typewriters, steam engines and unprogrammed computers, whose limitations can be illustrated by their inability to support a notion of *purpose.*

By contrast, a programmed computer may include representations of itself, its actions, possible futures, reasons for choosing, and methods of inference, and can therefore sometimes contain purposes which generate behaviour, as opposed to merely containing physical structures and processes which generate behaviour. So biologists and psychologists who aim to banish talk of purposes from science, thereby ignore some of the most important new developments in science. So do philosophers and psychologists who use the existence of purposive human behaviour to 'disprove' the possibility of a scientific study of man.

3. Learning that a computer contains a certain sub-program enables you to explain some of the things it can do, but provides no basis for predicting what it *always* or *frequently* does, since that will depend on a large number of other factors which determine when this sub-program is executed and the environment in which it is executed. So a scientific investigation of computational processes need not be primarily a search for *laws* so much as an attempt to describe and explain what sorts of things are and are not *possible.* A central form of question in science and philosophy is 'How is so and so possible?' Many scientists, especially those studying people and social systems, mislead themselves and their students into thinking that science is essentially a search for laws and correlations, so that they overlook the study of possibilities. Linguists (especially since Chomsky) have grasped this point, however. (This topic is developed at length in chapter 2.)

4. Similarly there is a wide-spread myth that the scientific study of complex systems requires the use of numerical measurements, equations, calculus, and the other mathematical paraphernalia of physics. These things are useless for describing or explaining the important aspects of the behaviour of complex programs (e.g. a computer, operating system, or Winograd's program described in his book *Understanding Natural Language*).

Instead of equations and the like, quite new non-numerical formalisms have evolved in the form of programming languages, along with a host of informal concepts relating the languages, the programs expressed therein, and the processes they generate. Many of these concepts (e.g. *parsing, compiling, interpreting, pointer, mutual recursion, side-effect, pattern matching)* are very general, and it is quite likely that they could be of much more use to students of biology, psychology and social science than the kinds of numerical mathematics they are normally taught, which are of limited use for theorising about complex interacting structures. Unfortunately although many scientists dimly grasp this point (e.g. when they compare the DNA molecule with a computer program) they are often unable to *use* the relationship: their conception of a computer program is limited to the sorts of data-processing programs written in low-level languages like Fortran or Basic.

5. It is important to distinguish cybernetics and so-called 'systems theory' from this broader science of computation, for the former are mostly concerned with processes involving relatively fixed structures in which something quantifiable (e.g. money, energy, electric current, the total population of a species) flows between or characterises substructures. Their formalisms and theories are too simple to say anything precise about the communication of a sentence, plan or problem, or to represent the process of construction or modification of a symbolic structure which stores information or abilities.

Similarly, the mathematical theory of information, of Shannon and Weaver, is mostly irrelevant, although computer programs are often said to be information-processing mechanisms. The use of the word 'information' in the mathematical theory has proved to be utterly misleading. It is not concerned with meaning or content or sense or connotation or denotation, but with probability and redundancy in

signals. If more suitable terminology had been chosen, then perhaps a horde of artists, composers, linguists, anthropologists, and even philosophers would not have been misled.

I am not denying the importance of the theory to electronic engineering and physics. In some contexts it is useful to think of communication as sending a signal down a noisy line, and understanding as involving some process of decoding signals. But human communication is quite different: we do not decode, we interpret, using enormous amounts of background knowledge and problem-solving abilities. That is, we map one class of structures (e.g. 2-D images), into another class (e.g. 3-D scenes). Chapter 9 elaborates on this, in describing work in computer vision. The same is true of artificial intelligence programs which understand language. Information theory is not concerned with such mappings.

6. One of the major new insights is that computational processes may be markedly decoupled from the physical processes of the underlying computer. Computers with quite different basic components and architecture may be equivalent in an important sense: a program which runs on one of them can be made to run on any other either by means of a second program which simulates the first computer on the second, or by means of a suitable compiler or interpreter program which *translates* the first program into a formalism which the second computer can execute. So a program may run on a *virtual* machine.

Differences in size can be got round by attaching peripheral storage devices such as magnetic discs or tapes, leaving only differences in speed.

So all modern digital computers are theoretically equivalent, and the detailed physical structure and properties of a computer need not constrain or determine the symbol-manipulating and problem-solving processes which can run on it: any constraints, except for speed, can be overcome by providing more storage and feeding in new programs. Similarly, the programs do not determine the computers on which they can run.

7. Thus reductionism is refuted. For instance, if biological processes are computational processes running on a physico-chemical computer, then essentially the same processes could, with suitable re-programming, run on a different sort of computer. Equally, the same computer could permit quite different computations: so the nature of the physical world need not determine biological processes. Just as the electronic engineers who build and maintain a computer may be quite unable to describe or understand some of the programs which run on it, so may physicists and chemists lack the resources to describe, explain or predict biological processes. Similarly psychology need not be reducible to physiology, nor social processes to psychological ones. To say that wholes may be more than the sum of their parts, and that qualitatively new processes may 'emerge' from old ones, now becomes an acceptable part of the science of computation, rather than old-fashioned mysticism. Many anti-reductionists have had this thought prior to the development of computing, but have been unable to give it a clear and indisputable foundation.

8. There need not be only *two* layers: programs and physical machine. A suitably programmed computer (e.g. a computer with a compiler program in it[2]), is itself a new computer a new 'virtual machine' which in turn may be programmed so as to support new kinds of processes. Thus a single process may involve many layers of computations, each using the next lower layer as its underlying machine. But that is not all. The relations may sometimes not even be hierarchically organised, for instance if process A forms part of the underlying machine for process B and process B forms part of the underlying machine for process A. Social and psychological, psychological and physiological processes, seem to be related in this mutually supportive way. Chapters 6 and 9 present some examples. The development of good tools for thinking about a system composed of multiple

interlocking processes is only just beginning. Systems of differential equations and the other tools of mathematical physics are worse than useless, for the attempt to use them can yield quite distorted descriptions of processes involving intelligent systems, and encourage us to ask unfruitful questions.

9. Philosophers sometimes claim that it is the business of philosophy only to analyse concepts, not to criticise them. But constructive criticism is often needed and in many cases the task will not be performed if philosophers shirk it. An important new task for philosophers is constructively critical analysis of the concepts and underlying presuppositions emerging from computer science and especially artificial intelligence. Further, by carefully analysing the mismatch between some of our very complicated ordinary concepts like *goal, decide, infer, perceive, emotion, believe, understand,* and the models being developed in artificial intelligence, philosophers may help to counteract unproductive exaggerated claims and pave the way for further developments. They will be rewarded by being helped with some of their philosophical problems.

10. For example, the computational metaphor, paradoxically, provides support for a claim that human decisions are not physically or physiologically determined, since, as explained above, if the mind is a computational process using the brain as a computer then it follows that the brain does not constrain the range of mental processes, any more than a computer constrains the set of algorithms that can run on it. It can be more illuminating to think of the program (or mind) as constraining the physical processes than vice versa.

Moreover, since the state of a computation can be frozen, and stored in some non-material medium such as a radio signal transmitted to a distant planet, and then restarted on a different computer, we see that the hitherto non-scientific hypothesis that people can survive bodily death, and be resurrected later on, acquires a new lease of life. Not that this version is likely to please theologians, since it no longer requires a god.

11. Recent attempts to give computers perceptual abilities seem to have settled the empiricist/rationalist debate by supporting Immanuel Kant's claim that no experiencing is possible without information-processing (analysis, comparison, interpretation of data) and that no information-processing is possible without pre-existing knowledge in the form of symbol-manipulating procedures, data-structures, and quite specific descriptive abilities. (This topic is elaborated in chapter 9.)

Shallow philosophical, linguistic and psychological disputes about innate or non-empirical knowledge are being replaced by much harder and deeper explorations of exactly what pre-existing knowledge is required, or sufficient, for particular types of empirical and non-empirical learning. What knowledge of two- and three-dimensional geometry and of physics does a robot need in order to be able to interpret its visual images in terms of tables, chairs and dishes to be carried to the sink? What kind of knowledge about its own symbolisms and symbol-manipulating procedures will a baby robot need in order to stumble upon and understand the discovery that counting a row of buttons from left to right necessarily produces the same result as counting from right to left, if no mistakes occur? (More on this sort of thing in the chapter on learning about numbers.)

Similarly, philosophical debates about the possibility of 'synthetic apriori' knowledge dissolve in the light of new insights into the enormous variety of ways in which a computational system (including a human society?) may make inferences, and perhaps discover necessary truths about the capabilities and limitations of its current stock of programs. For an example see the book by Sussman about a program which learns to build better programs for stacking blocks by analysing why initial versions go wrong.
(G.J. Sussman, *A Computational Model of Skill Acquisition,* American Elsevier, 1975.)

Epistemology, developmental psychology, and the history of ideas (including science and art) may be integrated in a single computational framework. The chapters on the aims of science and on number concepts are intended as a small step in this direction.

12. One of the bigger obstacles to progress in science and philosophy is often our inability to tell when we lack an explanation of something. Before Newton, people thought they understood why unsupported objects fell. Similarly, we think practice explains learning, familiarity explains recognition, desire explains action. Philosophers often assume that if you have experienced instances and non-instances of some concept, then this 'ostensive definition' suffices to explain how you could have learnt this concept. So our experience of seeing blue things and straight lines is supposed to explain how we acquire the concepts *blue* and *straight.* As for *how* the relevant aspects of instances and non-instances are noticed, related to one another and to previous experiences, and how the irrelevant aspects are left out of consideration the question isn't even asked. (Winston asked it, and gave some answers to it in the form of a primitive learning program: see his 1975.) Psychologists don't normally ask these questions either: having been indoctrinated with the paradigm of dependent and independent variables, they fail to distinguish a study of the *circumstances* in which some behaviour does and does not occur, from a search for an *explanation* of that behaviour.

People assume that if a person or animal wants something, then this, together with relevant beliefs, suffices to explain the resulting actions. But no decent theory is offered to explain *how* desires and beliefs are capable of generating action, and in particular no theory of how an individual finds relevant beliefs in his huge store of information, or how conflicting motives enter into the process, or how beliefs, purposes, skills, etc. are combined in the design of an action (e.g. an utterance) suited to the current situation. The closest thing to a theory in the minds of most people is the model of desires as physical forces pushing us in different directions, with the strongest force winning. The mathematical theory of games and decisions is a first crude attempt to improve on this, but is based on the false assumptions that people start with a well-defined set of alternative actions when they take decisions.

Work in artificial intelligence on programs which formulate and execute plans is beginning to unravel some of the intricacies of such processes. My chapter on aspects of the mechanism of mind will discuss some of the problems. (Chapter 6).

By trying to turn our explanations and theories into designs for working systems, we soon discover their poverty. The computer, unlike academic colleagues, is not convinced by fine prose, impressive looking diagrams or jargon, or even mathematical equations. If your theory doesn't work then the *behaviour* of the system you have designed will soon reveal the need for improvement. often errors in your design will prevent it behaving at all.

Books don't behave. We have long needed a medium for expressing theories about behaving systems. Now we have one, and a few years of programming explorations can resolve or clarify some issues which have survived centuries of disputation.

Progress in philosophy (and psychology) will now come from those who take seriously the attempt to *design a person.* I propose a new criterion for evaluating philosophical writings: could they help someone designing a mind, a language, a society or a world?

The same criterion is relevant to theorising in psychology. The difference is that philosophy is not so much concerned with finding the correct explanation of *actual* human behaviour. Its aims are more general. For more on the difference see chapters 2 and 3.

13. A frequently repeated discovery, using the new methodology, is that what seemed simple and easy to explain turns out to be very complex, requiring sophisticated computational resources, for instance: seeing a dot, remembering a word, learning from an example, improving through practice, recognising a familiar shape, associating two ideas, picking up a pencil. Of course, it may be that for all these achievements there are simple explanations, of kinds hitherto quite unknown. But at least we have learnt that we don't know them, and that is real progress. This also teaches a new respect for the intellects of infants and other animals. How does a bee manage to alight on a flower without crashing into it?

14. There are some interesting implications of the points made in 7 and 8 above. I mentioned that two computational processes may be mutually supportive. Similarly, two procedures may contain each other as parts, two information structures may contain each other as parts. More generally, a whole system may be built up from large numbers of *mutually recursive* procedures and data-structures, which interlock so tightly that no element can be properly defined except in terms of the whole system. (Recursive rules in formal grammars illustrate the same idea.) Since the system cannot be broken down hierarchically into parts, then parts of those parts, until relatively simple concepts and facts are reached, it follows that anyone learning about the system has to learn many different interrelated things in parallel, tolerating confusion, oversimplifications, inaccuracies, and constantly altering what has previously been learnt in the light of what comes later.[3]

So the process of *learning* a complex interlocking network of circular concepts, theories and procedures may have much in common with the task of *designing* one.

If all this is correct it not only undermines philosophical attempts to perform a logical analysis of our concepts in terms of ever more primitive ones (as Wittgenstein, for example, assumed possible in his *Tractatus Logico Philosophicus)*, it also has profound implications for the psychology of learning and for educational practice. It seems to imply that learning may be a highly creative process, that cumulative educational programmes may be misguided, and that teachers should not expect pupils to get things right while they are in the midst of learning a collection of mutually recursive concepts. This theme will be illustrated in more detail in the chapter on learning about numbers.

(One implication is that this book cannot be written in such a way as to introduce readers to the main ideas one at a time in a clear and accurate way. Readers who are new to the system of concepts will have to revisit different portions of the book frequently. No author has the right to expect this. The book is therefore quite likely to fail to communicate.)

15. Much of what is said in this book simply reports *common sense.* That is, it attempts to articulate much of the sound intuitive knowledge we have picked up over years of interacting with the physical world and with other people.

Making common sense explicit is the goal of much philosophising. Common sense should not be confused with *common opinions,* namely the beliefs we can readily formulate when asked: these are often false over-generalisations or merely the result of prejudice. Common sense is a rich and profound store of information, not about laws, but about what people are capable of doing, thinking or experiencing.

But common sense, like our knowledge of the grammar of our native language, is hard to get at and articulate, which is one reason why so much of philosophy, psychology and social science is vapid, or simply false.

Philosophers have been struggling for centuries to develop techniques for articulating common sense and unacknowledged presuppositions, such as the techniques of conceptual analysis and the exploration of paradoxes. Artificial intelligence provides an important new tool for doing this. It helps us find our mistakes quickly. One reason for this is that attempts to make computers understand what we say soon break down if we haven't learnt to articulate in the programs the presuppositions and rich conceptual structures which we use in understanding such things. (See Abelson, 'The structure of belief systems', and Schank & Abelson, 1977.)

Further, when you've designed a program whose behaviour is meant to exemplify some familiar concept, such as learning, perceiving, conversing, or achieving a goal, then in trying to interact with the program and in experiencing its behaviour it often happens that you come to realise that it does not really exemplify your concept after all, and this may help you to pin down features of the concept, essential to its use, which you had not previously noticed. So artificial intelligence contributes to conceptual analysis. (The interaction is two-way.)

16. Of course, merely *imagining* the program's behaviour would often suffice: doing the program isn't necessary in principle. But one of the sad and yet exhilarating facts most programmers soon learn is that it is hard to be sufficiently imaginative to anticipate the kinds of behaviour one's program can produce, especially when it is a complex system capable of generating millions of different kinds of processes depending on what you do with it. It is a myth that programs do just what the programmer intended them to do, especially when they are interacting with compilers, operating systems and hardware designed by someone else. The result is often behaviour that nobody planned and nobody can understand.

Thus new possibilities are discovered. Such discoveries may serve the same role as thought-experiments have often done in physics. So computational experiments may help to extend common sense as well as helping us to analyse it.

17. One of the things I have been trying to do is undermine the conflict between those who claim that a scientific study of man is possible and those who claim it isn't. Both sides are usually adopting a quite mistaken view of the essence of science. Bad philosophical ideas seem to have a habit of pervading a whole culture (like the supposed dichotomy between the emotional, intuitive aspects of people and the cognitive, intellectual, or rational aspects -- a dichotomy I have tried to undermine elsewhere).

The chapter on the aims of science attempts to correct widespread but mistaken views about the nature of science. I first became aware of the mistakes under the influence of linguistics and artificial intelligence.

18. One of the main themes of the revolution is that the pure scientist needs to behave like an engineer: designing and testing *working* theories. The more complex the processes studied, the closer the two must become. Pure and applied science merge. And philosophers need to join in.

19. I'll end with one more wildly speculative remark. Social systems are among the most complex computational processes created by man (whether intentionally or not). Most of the people currently charged with designing, maintaining, improving or even studying such processes are almost completely ignorant of the concepts, and untrained in the skills, required for thinking about very complex interacting processes. Instead they mess about with *variables* (on ordinal, interval or ratio scales), looking for *correlations* between them, convinced that *measurement* and *laws* are the stuff of science, without recognizing that such techniques are merely useful stop-gaps for dealing with phenomena you don't yet understand. In years to come, our willingness to trust these politicians, civil servants, economists, educationalists and the like with the task of managing our social system will look rather laughable. I am not suggesting that programmers should govern us. Rather, I venture to

suggest that if everyone were allowed to play with computers from childhood, not only would education become much more fun and stretch our minds much further, but people might be a lot better equipped to face many of the tasks which currently defeat us because we don't know how to think about them. Computer 'experts' would find it harder to exploit us.

1.4. What is Artificial Intelligence?

The best way to answer this question is to look at the aims of A.I., and some of the methods for achieving those aims, and to show how the subject is decomposable into sub-domains and related to other disciplines. This would require a whole book, which is not my current purpose. So I'll give an incomplete answer by describing and commenting on some of the aims. AI is not just the attempt to make machines do things which when done by people are called ''intelligent''. It is much broader and deeper than this. For it includes the scientific and philosophical aims of *understanding* as well as the engineering aim of *making*.

The aims of Artificial Intelligence

1. Theoretical analysis of possible effective explanations of intelligent behaviour.
2. Explaining human abilities.
3. Construction of intelligent artefacts.

Comments on the aims:

a) The first aim is very close to the aims of Philosophy. The main difference is the requirement that explanations be 'effective'. That is they should form part of, or be capable of contributing usefully to the design of, a working system, i.e. one which generates the behaviour to be explained.

b) The second aim is often formulated, by people working in A.I., as the aim of designing machines which 'simulate' human behaviour, i.e. behave like people. There are many problems about this, e.g. which people? People differ enormously. Also what does like' mean? Programs, mechanisms, and people may be compared at many different levels.

c) The programming of computers is not an essential part of the first two aims: rather it is a research method. It imposes a discipline, and provides a tool for finding out what your explanations are theoretically capable of explaining. Sometimes they can do more than you intended usually less.

d) People doing A.I. do not usually bother much about experiments or surveys of the kinds psychologists and social scientists do, because the main current need is not for more *data* but for better theories and theory-building concepts and formalisms, so that we can begin to explain the masses of data we already have. (In fact a typical strategy for getting theory-building off the ground, in A.I. as in other sciences, is to try to explain idealised and simplified situations, in which much of the available data are ignored: e.g. A.I. programs concerned with 'toy' worlds (like the world of overlapping letters described in chapter 9), and physicists treating moving objects as point masses.)

e) An issue which bothers psychologists is how we can tell whether a particular program really does explain some human ability, as opposed to merely mimicking it. The short answer is that there is never any way of establishing that a scientific explanation is correct. However, it is possible to compare rival explanations, and to tell whether we are making progress. Criteria for doing this are formulated in chapter 2.

f) The notion of 'intelligent behaviour' in the first aim is easy to illustrate but hard to define. It includes behaviour based on the ability to cope in a systematic fashion with a range of problems of varying structures, and the ability (consciously or unconsciously) to build, describe, interpret, compare, modify and use complex structures, including symbolic structures like sentences, pictures, maps and plans for action. A.I. is not specially concerned with unusual or meritorious forms of intelligence: ordinary human beings and other animals display the kinds of intelligence whose possibility A.I. seeks to explain.

g) It turns out that there is not just one thing called 'intelligence', but an enormous variety of kinds of expertise the ability to see various kinds of things, the ability to understand a language, the ability to learn different kinds of things, the ability to make plans, to test plans, to solve problems, to monitor our actions, etc. It also includes the ability to have motives, emotions, and attitudes, e.g. to feel lonely, embarrassed, proud, disgusted, elated, and so on. Each of these abilities involves domain-specific knowledge (factual and procedural knowing that and knowing how). So, much current work in A.I. is exploration of the *knowledge* underlying competence in a variety of specialised domains seeing blocks, understanding children's stories, making plans for building things out of blocks, assembling bits of machinery, reading handwriting, synthesising or checking computer programs, solving puzzles, playing chess and other games, solving geometrical problems, proving logical and mathematical theorems, etc.

I.e. a great deal of A.I. research is highly 'domain-specific', and amounts to an attempt to explicitly formulate knowledge people already use unconsciously in ordinary life or specialised activities. This is closely related to conceptual analysis as practised by linguists and philosophers. (See Chapter 4.)

h) Alongside all this, there is the search for generality. So research is in progress on possible computing mechanisms and concepts which are not necessarily relevant only to one domain, but may be useful, or necessary, for explaining many different varieties of intelligence, e.g. mechanisms concerned with good ways of storing and retrieving information, making inferences, controlling processes, allowing sub-processes to interact and influence one another, allowing factual knowledge to be translated into procedural forms as required, etc. However, the role of general mechanisms seems to be much less important in explaining intelligent abilities than the role of domain specific knowledge.

i) As pointed out below, much of the domain-specific research overlaps with research in other disciplines, e.g. Linguistics, Psychology, Education, Philosophy, Anthropology, and perhaps Physiology. For example, you can't make a computer understand English without studying syntactic, semantic and pragmatic rules of English, that is, without doing Linguistics.

j) A major effect of A.I. research as already mentioned is to establish that apparently simple tasks, like seeing a line, may involve very complex cognitive processes, using substantial prior knowledge.

k) One side-effect of attempts to understand human abilities well enough to give them to computers, has been the introduction of some new approaches to teaching those abilities to children, for instance LOGO projects (see papers by Papert). These projects use a programming language based on programming languages developed for A.I. research, and they teach children and other beginners programming using such a language. These languages are much more suitable for teaching beginners than BASIC or FORTRAN, the most commonly used languages, because (a) they are very much more powerful, making it relatively easy to get the computer to do complex things and (b) they are not restricted to numerical computations. For example, LOGO, used at MIT and Edinburgh University, and POP-2, which we use at Sussex University, provide facilities suitable for manipulating words and sentences, drawing pictures, etc. (See Burstall et al. 1971.)

l) A.I. gives people much more respect for the achievements of children, and more insight into the problems they have to solve in learning what they do. This leads to a better understanding of possible reasons for not learning so well.

1.5. Conclusion

The primary aim of my research is to understand aspects of the human mind. Different people will be interested in different aspects, and many will not be interested in the aspects I have chosen: scientific creativity, decision making, visual perception, the use of verbal and non-verbal symbolisms, and learning of elementary mathematics. At present I can only report fragmentary progress. Whether it is called philosophy, psychology, computing science, or anything else doesn't really interest me. The methods of all these disciplines are needed if progress is to be made. It may be that the human mind is too complex to be understood by the human mind. But the desire to attempt the impossible seems to be one of its persistent features.

Note

The remaining chapters, apart from chapter 10 should be readable in any order. On the whole, people knowledgeable about philosophy and ignorant of computing will probably find chapters 2 to 5 easier than the following chapters. People interested in trying to understand how people work, and not so concerned with abstract methodological issues, may find chapters 2 to 5 tedious (or difficult?), and should start with Part Two, though they'll not be able to follow all the methodological asides, which refer back to earlier chapters.

Endnotes

(1) I write 'program' not 'programme' since the former is a technical term referring to a collection of definitions, instructions and information expressed in a precise language capable of being interpreted by a computer. For more details see J. Weizenbaum, *Computer Power and Human Reason.* There is much in this book that I disagree with, but it is well worth reading, and may be a useful antidote to some of my excesses.

(2) A compiler is a program which translates programs from one programming language into another. E.g. an ALGOL compiler may translate ALGOL programs into the 'machine code' of a particular computer.

(3) Apparently Hegel anticipated some of these ideas. His admirers might advance their understanding of his problems by turning to the study of computation.

Last updated: 4 Jun 2007; reformatted 1 Jul 2015