16

A computational analysis of consciousness

P. N. Johnson-Laird

This chapter argues that the problems of consciousness are only likely to be solved by adopting a computational approach and by setting a number of tractable goals for the theory. Four such phenomena need to be explained: the division between conscious states and unconscious mental processes; the relative lack of conscious control over many emotions and behaviours; the unique subjective experience of self-awareness; and that aspect of intentions that is missing from goal-directed computer programs and automata. The chapter outlines a theory of consciousness based on three main assumptions: the computational architecture of the mind consists in a hierarchy of parallel processors; the processor at the top of the hierarchy is the source of conscious experience; this processor—the operating system—has access to a model of itself, and the ability to embed models within models recursively. If the thesis of the chapter is correct, the four problems of consciousness can be solved once we understand what it means for a computer program to have a high-level model of its own operations.

The problems of consciousness

What should a theory of consciousness explain? This is perhaps the first puzzle about consciousness, because unlike, say, the mechanism of inheritance, it is not clear what needs to be accounted for. One might suppose that a theory should explain what consciousness is and how it can have particular objects; that is, the theory should account for the qualitative aspects of the phenomenal experience of awareness. The trouble is that there are no obvious criteria by which to assess such a theory. Indeed, this lack of criteria lent respectability to the behaviouristic doctrine that consciousness is not amenable to scientific investigation; i.e. that it is a myth that the proper study of nerve, muscle, and behaviour will ultimately dispel. A prudent strategy is

therefore both to take a different approach to consciousness and to suggest a more tractable set of problems for the theory to solve. My approach here will be to assume that consciousness is a computational matter that depends on how the brain carries out certain computations, not on its physical constitution. I will outline four principal problems that a theory of consciousness should solve. I will then propose a theory of mental architecture—a conjecture about how the mind is organized —and, finally, I will show that it could provide answers to these problems.

1. *The problem of awareness.* When someone speaks to you, you can be aware of the words they utter, you can be aware of the meaning of their remarks, and you can be aware of understanding what they are saying. And if none of this information is available to you, then you can be aware of that, too. Yet there is much that is permanently unavailable to you. You cannot be aware of how you understand the speaker's remarks or of the form in which their meaning is represented in your mind. In fact, you can never be completely conscious of how you exercise any mental skill. A theory of consciousness must explain how it is possible for there to be this division between conscious states and unconscious processes.

2. *The problem of control.* You cannot consciously control all of your feelings. You can feign happiness and sadness, but it is difficult, if not impossible, for you to evoke these emotions merely by a conscious decision: your best hope is to use a strategy, such as thinking of a particular situation that evoked the relevant emotion, but even this method may fail. Conversely, a particular feeling may overwhelm you despite all your efforts to resist it. This lack of control extends, of course, to behaviour. You may, for instance, intend to give up smoking but be unable to stick to your intention. Some individuals can exert a tight control on themselves and on their expressions of emotion; others, as Oscar Wilde said of himself, can resist everything except temptation. There do indeed seem to be differences in will-power from one individual to another, though the topic is almost taboo in cognitive psychology.

3. *The problem of self-awareness.* You can be aware of the task that you are carrying out and be so absorbed in it as to forget yourself. Alternatively, you can be self-aware and be conscious of what you yourself are currently doing. Sometimes, perhaps, you can even be aware that you are aware that you are aware ... and so on. Self-awareness is, of course, essential for a sense of one's integrity, continuity, and individuality.

4. *The problem of intentions.* To act intentionally is at the very least to decide to do something (for some reason or to achieve some goal) and then in consequence to do it. There are many computer programs, however, that are governed by internally generated goals; for example, the programming language PLANNER enables programs to be written that set up goals, and that then seek to achieve those goals by simulating the action of a non-deterministic automaton (see Hewitt 1972). These programs may provide a reasonable account of unconscious intentions —goals that influence your behaviour without your conscious realization, but the programs do not have your capacity to formulate conscious intentions, because they have no awareness of what they are doing (see Marcel 1988, this volume). A theory of consciousness should elucidate the component of intentional behaviour that is missing from such programs. Metacognition is similarly missing from programs: you can, for example, reason about how you carry out some task, such as reasoning itself, and then use the results of such cogitations to modify your performance. Programs can carry out many tasks but they do not yet reason about their own performance. As we shall see, self-awareness, intentions, and metacognition all appear to call for the same computational mechanism.

Psychologists and others have, of course, proposed theories of consciousness. They have tried to account for it in terms of the evolution of more complex brains (e.g. John 1976), or of more complex behaviours culminating in linguistic communication and social relations (e.g. Mead 1934). But consciousness is unlikely to be merely a consequence of more neurones with more connections between them; it almost certainly depends on how the neurons are connected and the nature of the computations they carry out. Likewise, if language and society could have evolved without consciousness, why should they need it, and how would they be able to awaken our slumbering minds? Psychologists have also identified consciousness with the contents of a limited capacity processing mechanism (Posner and Boies 1971), with a device that determines what actions to take and what goals to seek (Shallice 1972), and with a particular mode of information processing that affects the mental structures governing actions (Mandler 1975). These claims are plausible, but they were not addressed to the four problems above, and they do not solve them. They account for some of the characteristics of consciousness, but they might well apply to a device such as a computer running a PLANNER-like program. My aim is to sketch a computational approach to consciousness that may lead to solutions to the unsolved problems of awareness, control, self-awareness, and intentional behaviour.

Hierarchical parallel processing

From the simple nerve networks of coelenterates to the intricacies of the human brain, there appears to be a uniform computational principle: asynchronous parallel processing. That mental processes occur in parallel is also borne out by the fact that, for example, language is organized at different levels—speech sounds, morphemes, sentences, and discourse—and processed at these levels contemporaneously. There are good reasons to suppose that one processor in the parallel system cannot directly modify the internal instructions of another, because such interactions—even if they were physically possible—would produce highly unstable and unpredictable consequences. A more plausible form of interaction relies solely on the communication of messages between the processors. These messages may take the form of predictions, constraints on processing, the results of computations, emergency signals, and other such interrupts.

The most general conception of a system of parallel processing is of a set of finite-state automata that have channels between them for communicating data, and that operate serially and according to the asynchronous principle that each processor starts to compute as soon as it receives the data that it needs. Other parallel systems are special cases of this design; for example, 'connectionist' systems in which only information about level of activation is passed from one processor to another (e.g. see Anderson and Hinton 1981; Rumelhart and McClelland 1986), systems in which all the processors are synchronized by reference to an internal clock (see Kung 1980), and vector machines in which all the processors carry out the same procedure (see Kozdrowicki and Theis 1980). It is important to keep in mind the distinction between a function and an algorithm for computing that function, because there are infinitely many different algorithms for computing any computable function (e.g. see Rogers 1967). Moreover, although any function that can be computed in parallel can be computed by a serial device, there are many algorithms that run on parallel computers that cannot run on serial ones. Hence if consciousness depends on the computations of the nervous system, then it is likely to be a property of the algorithms that are used to carry out those computations rather than a property of their results (Johnson-Laird 1983*a*): it ain't what you do, it's the way that you do it! One unfortunate consequence of this consideration is the lack of any characteristic hallmark in behaviour indicating that an organism is conscious. Some psychologists might at this point abandon the study of consciousness on the grounds that accounts of it are unlikely to have testable consequences in behaviour. Such empirical pedantry may be premature: the development of computational accounts may lead to other forms of testing.

There are some problems—the parsing of certain abstract languages, for example—that can be shown to be solvable in principle but not in practice: they are to computation what Malthus's doctrine of population growth is to civilization. A problem is inherently intractable when any algorithm for it takes a time that grows exponentially with the size of the input (e.g. see Hopcroft and Ullman 1979). For an input of n, where n is small, an algorithm may be feasible, but even if the time it takes is proportional, say, to 2^n, then, because such exponentials increase at so great a rate with an increase of n, a computer the size of the universe operating at the speed of light would take billions of years to compute an output for a relatively modest input. Parallel processing is of no avail for rendering such problems tractable. What it does is to speed up the execution of algorithms that take only a time proportional to a polynomial of the size of the input. If many processors compute in parallel, they can divide up the task between them whenever there are no dependencies between the computations. Such a division of labour not only speeds up performance, but it also allows several processors to perform the same sub-task so that should one of them fail the effects will not be disastrous, and it enables separate groups of processors to specialize in different sub-tasks. The resulting speed, reliability, and specialization have obvious evolutionary advantages.

But parallel computation has its dangers too. One processor may be waiting for data from a second processor before it can begin to compute, but the second processor may itself be waiting for data from the first. The two processors will thus be locked in a 'deadly embrace' from which neither can escape. Any simple nervous system with a 'wired in' program that behaved in this way would soon be eliminated by natural selection. Higher organisms, however, can develop new programs—they can learn—and therefore there must be mechanisms, other than those of direct selective pressure, to deal with pathological configurations that may arise between the processors. A sensible design is to promote one processor to monitor the operations of others and to override them in the event of deadlocks and other pathological states of affairs. If this design feature is replicated on a large scale, the resulting architecture is an hierarchical system of parallel processors: a high-level processor that monitors lower level processors, which in turn monitor the processors at a still lower level, and so on down to the lowest level of processors governing sensory and motor interactions with the world. A hierarchical organization of the nervous system has indeed been urged by neuroscientists from Hughlings Jackson to H. J. Jerison (for the history of this idea see Oatley 1978), and Simon (1969) has argued independently that it is an essential feature of intelligent organisms.

The operating system

In a simple hierarchical computational device, the highest level of processing could consist of an operating system. The operating system of a digital computer is a suite of programs that allows a human operator to control the computer. There are instructions that enable the operator to recover a program stored on a magnetic disk, to compile it, to run it, to print out its source code, and so on. When the computer is switched on, its resident monitor is arranged to load the operating system either automatically or as a result of some simple instructions. The notion that the mind has an operating system verges, as we shall see, on the paradoxical, but it has some relatively straightforward consequences. The operating system must have considerable autonomy, though it must also be responsive to demands from other processors. It must be switched on and off by the mechanisms controlling sleep, though other processors continue to function. It must depend on a second level of processors for passing down more detailed control instructions to still lower levels, and for passing up interpreted sensory information. Doubtless, there are interactions between processors at the same or different levels, and facilities that allow priority messages from a lower level to interrupt computations at a higher level. However, the operating system does not have complete control over the performance of the hierarchy. In particular, as Oatley and I have argued (Oatley and Johnson-Laird 1987), emotional signals constitute a separate channel of communication in the hierarchy and set processors into characteristic modes that predispose the organism towards particular classes of behaviour. Conflicts within the hierarchy between different emotional modes are resolved, not by the operating system alone, but by some general architectural principle such as lateral inhibition.

The hierarchy of communicating parallel processors imposes one great virtue on the operating system: it can be relatively simple, because it does not need to be concerned with the detailed implementation of the instructions that it sends to lower level processors. It specifies what they have to do (e.g. to walk, to think, to talk) but not how they are to carry out the computations that underlie these tasks. It receives information from the lower processors about the results of computations, but not about how they were obtained. Thus vision makes explicit to the operating system *what* is *where* in the scene before us: we have little or no access to the sequence of representations that vision must depend on (Marr 1982). The visual world is presented in a way that is as real as the stone that Dr Johnson kicked in order to refute idealism. This phenomenal reality—our direct awareness of things—is a triumph of the adaptive nature of the mind. If we were aware that the visual world is a

representation, then we would be more likely to doubt its veridicality and to treat it as something to be pondered over—a potentially fatal debility in the event of danger. Psychology is difficult just because there is an evolutionary advantage in a seemingly direct contact with the world and in hiding the cognitive machinery from consciousness.

Some empirical evidence

Let us take stock of the theory so far. The brain is a parallel computer that is organized hierarchically. Its operating system corresponds to consciousness and it receives only the results of the computations of the rest of the system. Such a system can begin to account for the division between conscious and unconscious processes (the problem of awareness) and it can also allow the lower level processors a degree of autonomy (the problem of control). There are at least three clinical syndromes that corroborate this division. First, there is the phenomenon of 'blindsight' described by Weiskrantz *et al.* (1974). After damage to the visual cortex, certain patients report that they are blind in parts of the visual field, and their blindness is apparently confirmed by clinical tests. Yet, more subtle testing shows that the patients are able to use information from the 'blind' part of the field. It seems that their sight in the affected regions has continued to function but no longer yields an output to the operating system: they see without being conscious of what they see. Second, there are the 'automatisms' that occur after epileptic attacks. In this state, patients seem to function completely without consciousness and without the ability to make high-level decisions. They may be capable of driving a car, for example, but unable to respond correctly to traffic lights (Penfield 1975). Evidently, the attack leads to a dissociation between the operating system and the multiple processors. Third, there are the well-attested cases of hysterical paralysis. Prolonged stress may lead to paralyses that have, unbeknownst to the patient, no neurological explanation. They can often be cured, as the late Lord Adrian showed during World War I, by similarly duping the patient into believing that electrical stimuli will produce a cure (see Adrian and Yealland 1917). Since these patients are not malingering, they provide us with clear examples of a reaction that is outside the knowledge and control of the operating system.

Why is it that the contents of the operating system—more precisely, its working memory—are conscious, but all else in the hierarchy of processors is unconscious? In other words, what is it about the operating system that gives rise to the subjective experience of awareness? Of course, it is logically possible that each processor is fully aware but

cannot communicate its awareness to others—an analogous view was defended by William James in order to explain the phenomena of hysteria without having to postulate an unconscious mind. The view that I shall defend, however, is that the operating system's potential capacity for self-awareness is what gives rise to consciousness. Whenever any computational device is able to assess how it itself stands in relation to some state of affairs, it is—according to my hypothesis—conscious of that state of affairs. What needs to be elucidated is therefore how a computational device could assess its own relation to some state of affairs. It is to this problem that I now turn.

Self-awareness and the embedding of models

Reflection on the human capacity for self-reflection leads inevitably to the following observation: you can be aware of yourself. You also understand yourself to some extent, and you understand that you understand yourself, and so on. . . . The idea is central to the subjective experience of consciousness, yet it seems as paradoxical as the conundrum of an inclusive map. [If a large map of England were traced out in accurate detail in the middle of Salisbury Plain, then it should contain a representation of itself within the portion of the map depicting Salisbury Plain [which in turn should contain a representation of itself [which in turn should contain a representation of itself [and so on *ad infinitum*]]]]. Such a map is impossible because an infinite regress cannot occur in a physical object. Leibniz dismissed Locke's theory of the mind because there was just such a regress within it. However, a computational procedure for representing a map can easily be contrived to call itself recursively and thus to go on drawing the map within itself on an ever diminishing scale. The procedure could in principle run for ever: the values of the variables, though too small to be physically represented in a drawing, would go on diminishing perpetually.

There is a similar computational solution to the paradox of self-awareness. Ordinarily, when you perceive the world, vision delivers to your operating system a model that makes explicit the locations and identities of the objects in the scene [Marr 1982]. The operating system, however, can call on procedures that construct a model that makes explicit that it itself is perceiving the world: the contents of its working memory now contain a model representing it perceiving the particular state of affairs represented by the model constructed by perception. In other words, the visual model is embedded within a *model* of the operating system's current operation. Should you be aware that you are aware that you are perceiving the world, then there is a further embed-

ding: the operating system's working memory contains a model of it perceiving the state of affairs represented by the model of it perceiving the world. Since the hierarchy of embedded models exists simultaneously, the operating system can be aware that it is aware of the world. Granted the limited processing capacity of the operating system and its working memory, there is no danger of an infinite regress.

In self-awareness, there is a need for an element in the model of the current state of affairs to refer to the system itself and to be known so to refer. In the formation of intentions and metacognitions, there is a more complex requirement. To have a conscious intention, for instance, the operating system must elicit a representation of a possible state of affairs, and decide that it itself should act so as to try to bring about that state of affairs. An essential part of this process is precisely an awareness that the system itself is able to make such decisions. The system has to be able to represent the fact that the system can generate a representation of a state of affairs, and decide to work towards bringing it about. At a low level, there is a program (perhaps analogous to a program in PLANNER) that can construct a model of a state of affairs, and act so as to try to achieve it. (That is all that would be necessary for unconscious intentions). But the system can construct a *model* of itself operating at this low level of performance, and it can use this model in the process of making a decision. It can also construct a model of its own performance at this level in turn, and so on . . . to any required degree of embedding. Hence conscious intentions depend on having access to an element, not merely that refers to the system itself, but that represents the specific abilities of the system, and in particular its ability to plan and to act to achieve plans.

Metacognitive abilities similarly depend on access to such *models* of the system's capabilities, predilections, and preferences. You can reason about how you reason because you have access to a model of your reasoning performance. You can even reason about your metacognitive abilities—you think about how you tend, say, to concentrate too much on your past failures in trying to solve problems when you reflect on your reasoning performance. Thus, once again, you have the ability to make recursive embeddings of mental models within mental models. (I have argued elsewhere that this same ability underlies the phenomena of free will; see Johnson-Laird, in press). The recursive aspect of this ability is hardly problematical. The crux of the problem of consciousness resides in the other requirement: the operating system must have a partial model of itself in order to begin the process of recursion underlying intentionality.

No one knows what it means to say that an automaton or computer program has a model of itself. The question has seldom been raised and

certainly has yet to be answered. The notion must not be confused with self-description (*pace* Minsky 1968). It is a relatively straightforward matter to devise an automaton that can print out its own description (e.g. see Thatcher 1963). But such an automaton merely advertises its own inner structure in a way that is useful for self-reproduction, and it no more understands that description than a molecule of DNA understands genetics. A program might be devised, as Minsky (1968) argued, to use its self-description to predict its own future behaviour. Human beings, however, certainly do not have access to complete descriptions of themselves—if they did, any psychological problem could be solved by introspection. What is therefore needed is a program that has a *model* of its own high-level capabilities. This model would be necessarily incomplete, according to the present theory, and it might also be slightly inaccurate, but it would none the less be extremely useful. People do indeed know much about their own high-level capabilities: their capacity to perceive, remember, and act; their mastery of this or that intellectual or physical skill; their imaginative and ratiocinative abilities. They obviously have access only to an incomplete model, which contains no information about the inner workings of the web of parallel processors. It is a model of the major options available to the operating system.

Conclusions

The present approach to consciousness depends on putting together the three main components that I have outlined; hierarchical parallel processing, the recursive embedding of models, and the high-level model of the system itself. Self-awareness depends on a recursive embedding of models containing tokens denoting the self so that the different embeddings are accessible in parallel to the operating system. Metacognitions and conscious intentions depend on a recursive embedding of a model of elements of the self within itself, and of course the ability to use the resulting representation in thought.

This approach assumes that human behaviour depends on the computations of the nervous system. The class of procedures that I have invoked are, with the exception of a program that has a high-level model of itself, reasonably well understood. The immediate priority is therefore to attempt to construct such a program. It is often said that the computer is merely the latest in a long line of inventions—wax tablets, clockwork, steam engines, telephone switchboards—that have been taken as metaphors for the brain. What is often overlooked is that no one has yet succeeded in refuting the thesis that any explicit description of

an algorithm is computable. If that thesis is true, then all that needs to be discovered is what functions the brain computes and how it computes them. The computer is the last metaphor for the mind.

Acknowledgements

This chapter is a revised version of a paper that appeared in *Cognition and Brain Theory* (Johnson-Laird 1983*b*). I am grateful to the editors of this book for their forbearance and for their helpful suggestions about how the paper could be improved. I am also indebted to Carlo Umiltà and Alan Allport for their constructive criticisms of the earlier version of the paper.

References

Adrian, E. D. and Yealland, L. R. (1919). The treatment of some common war neuroses. *Lancet*, June, 3–24.

Anderson, J. A. and Hinton, G. E. (1981). Models of information processing in the brain. In *Parallel models of associative memory*, (ed. G. E. Hinton and J. A. Anderson), pp. 9–48. Lawrence Erlbaum Associates, Hillsdale, NJ.

Hewitt, C. (1972). *Description and theoretical analysis (using schemata) of PLANNER*. Memorandum AI TR-258. MIT Artificial Intelligence Laboratory.

Hopcroft, J. E. and Ullman, J. D. (1979). *Formal languages and their relation to automata*. Addison-Wesley, Reading, MA.

John, E. R. (1976). A model of consciousness. In *Consciousness and self-regulation: advances in research, Vol. 1*, (ed. G. E. Schwartz and D. Shapiro), pp. 21–51. John Wiley, Chichester, Sussex.

Johnson-Laird, P. N. (1983*a*). *Mental models*. Cambridge University Press, Cambridge, MA.

Johnson-Laird, P. N. (1983*b*). A computational analysis of consciousness. *Cognition and Brain Theory*, **6**, 499–508.

Johnson-Laird, P. N. (in press). Freedom and constraint in creativity. In *Creativity*, (ed. R. J. Sternberg). Cambridge University Press.

Kozdrowicki, E. W. and Theis, D. J. (1980). Second generation of vector supercomputers. *Computer*, **13**, 71–83.

Kung, H. T. (1980). The structure of parallel algorithms. In *Advances in computers*, Vol. 19, (ed. M. C. Yovits), pp. 53–73. Academic Press, New York.

Mandler, G. (1975). *Mind and emotion*. John Wiley, New York.

Marcel, A. J. (1988). Phenomenal experience and functionalism. In *Consciousness in contemporary science*, (ed. A. J. Marcel and E. Bisiach), p. 121. Oxford University Press.

Marr, D. (1982). *Vision: a computational investigation in the human representation of visual information.* Freeman, San Francisco.

Mead, G. H. (1934). *Mind, self and society—from the standpoint of a social behaviorist.* (ed. C. W. Morris),. University of Chicago Press.

Minsky, M. L. (1968). Matter, mind, and models. In *Semantic information processing,* (ed. M. L. Minsky), pp. 425–32. MIT Press, Cambridge, MA.

Oatley, K. (1978). *Perceptions and representations: the theoretical bases of brain research and psychology.* Methuen, Andover, Hants.

Oatley, K. and Johnson-Laird, P. N. (1987). Towards a cognitive theory of emotions. *Cognition and Emotion,* **1**, 3–28.

Penfield, W. (1975). *The mystery of mind.* Princeton University Press.

Posner, M. I. and Boies, S. J. (1971). Components of attention. *Psychological Review,* **78**, 391–408.

Rogers, H. (1969). *Theory of recursive functions and effective computability.* McGraw-Hill, New York.

Rumelhart, D. E. and McClelland, J. L. (ed.) (1986). *Parallel distributed processing: explorations in the microstructure of cognition. Vol. 1.: Foundations.* MIT Press, Cambridge, MA.

Shallice, T. (1972). Dual functions of consciousness. *Psychological Review,* **79**, 383–93.

Simon, H. A. (1969). *The sciences of the artificial.* MIT Press, Cambridge, MA.

Thatcher, J. W. (1963). The construction of a self-describing Turing machine. In *Mathematical theory of automata,* Microwave Research Institute Symposia, Vol. 12, (ed. J. Fox), pp. 165–71. Polytechnic Press, Polytechnic Institute of Brooklyn, New York.

Weiskrantz, L., Warrington, E. K., Sanders, M. D., and Marshall, J. (1974). Visual capacity in the hemianopic field following a restricted occipital ablation. *Brain,* **97**, 709–28.