MACHINES WITH SENSE
COMPUTERS AND PROCESSES OF THOUGHT

J. C. PÉNOCHET*

trans. Quinn FOERCH

A few centuries ago, the subtitle of this presentation might well have been: "The Catapult and Thought Processes." There is no shortage of historical interpretations regarding the workings of the human brain: for the Greeks, it seems, one could plausibly attribute to it the characteristics of a catapult. In the work of Leibniz, its function is likened to that of a mill. Sherrington, the great British neurophysiologist, compared it to a telegraph system. And during the childhood of the oldest among us, the analogy to a massive telephone exchange did not seem at all far-fetched. Today, to what extent is it a serious proposition to characterize it as a highly powerful digital computer?

Brain or machine—or the brain as a machine—this debate is hardly new. If the brain is the seat of thought, and if thought is a form of computation, can a machine not compute just as effectively? What relationship might digital computers bear to the nature of thought? Can "machines" actually "think"? Although this question has already been the subject of well over a thousand publications, it appears—at least until now—to have sparked little interest among psychiatrists and psychoanalysts, who tend to be less drawn to the realm of binary logic—and, no doubt, overly confident in the complexity of the human psyche to imagine that a machine could ever equal it. They are also more attached to the affective dimension than to pure intellectual performance. However, the progress achieved in the fields of artificial intelligence, knowledge engineering, and expert systems lends this debate an unprecedented prominence. At various levels, collaboration between computer scientists and specialists in psychological functioning proves highly desirable. My presentation will be limited to summarizing a few basic concepts regarding what is meant by "artificial intelligence," leaving it to other speakers—during these proceedings or at a later date—to elaborate upon or critique specific aspects of this broad debate.

A first section will review the concepts essential to understanding the possibilities and limitations of computing. The relationship to thought processes constitutes the very subject matter of cognitive psychology. In a second section, we will need to define the project itself and its epistemological status. Finally—and while we wait to discover what "computing machines" think of us—I will offer a few reflections on what humans tend to think of their own reflective capabilities; in short, I will speak to you about "strange loops."

The Limits of Computing

*Psychoanalyst, Hospital Psychiatrist, La Colombière Hospital – Montpellier University Hospital (CHU), 34059 Montpellier Cedex

Computer systems consist of both hardware and software. Alan Kay (4) offers us a beautiful definition: computers are to calculation what instruments are to music. Software is the musical score, the interpretation of which amplifies our knowledge and elevates our spirit. The materials of computing serve as extraordinarily dense means for the encoding and storage of information. And all information—for Gregory Bateson just as for Jacques Lacan—comprises both the physical form of the message (whether the hieroglyphs on clay tablets or the binary digits on magnetic media) and its interpretation. The entire question lies in determining to what extent the computer that "processes" the data is capable of "interpreting" it.

Hardware Limitations

There is no need to dwell here on the computational methods used for reading, processing, and storing messages—just as, in meteorology, one does not concern oneself with the movement of electrons; or as, in psychoanalysis, one pays no heed to the physical characteristics of the sound waves produced by the patient. The demonstration proceeds without them. For provided one possesses sufficient capacity to store encoded information, a very simple set of instructions suffices to represent any machine capable of processing information—even if the task involves simulating a completely novel computer. This was something Lady Ada Lovelace had already grasped perfectly when she programmed the universal Analytical Engine designed by Charles Babbage. Later, the so-called Turing machine served to firmly establish the concept that any computer can simulate another—whether that other machine already exists or is yet to be created. This is, of course, the philosophical consequence of such an assertion—one that must be borne in mind—is that a simulation of a machine can never be equivalent to the machine itself. And on a practical level, we certainly wish to obtain results within a timeframe of less than half a million years! We are thus building machines that, while remaining fundamentally similar, are also becoming ever more powerful—succeeding, to date, in pushing the boundaries of their physical substrates.

However, one essential fact demands our attention: while these machines operate on the same principles as their predecessors, significant quantitative modifications to their computational capacity—driven by increased execution speeds—trigger changes of scale. And these changes of scale ultimately manifest to us as qualitative modifications. For instance, its processing speed enables the binary computer to manipulate complex symbolic constructs just as effectively as it handles numbers. By virtue of its symbolic nature, the message produced becomes emancipated from the material media through which it is generated—much in the same way that a digital audio disc contains all the information necessary to evoke in us the emotional experience of a Bach fugue. To be precise: the computer is capable of processing any category of symbols to which human beings ascribe meaning.

Software Limitations

We are all familiar with the cliché of the seven-year-old child who learns to use a computer in a matter of hours, while his parents look on in wonder at having brought such a genius into the world. One is then left to wonder why software takes so long to develop, costs so much, and is—more often than not—riddled with "bugs." The answer is quite simple: computers are eminently logical (though neither stupid nor narrow-minded); being obedient by nature, they persist in doing exactly what they are told to do, rather than what we wish them to do. Writing a program amounts to drafting a sequential list of operations that the machine is required to execute, one after the other. Since the machine possesses no faculty of discrimination, the program must be devoid of ambiguity; this rules out natural languages and necessitates the highly rigid syntax characteristic of programming languages.

The Transformational Aspect of Programming

Barred from producing metaphorical or metonymic effects, programming languages suffer from the fundamental flaw of the signifier—namely, the ultimate inability to signify. One of the greatest difficulties in programming stems from its transformational nature—that is, from the fact that the program, in reality, says nothing about the intermediate situations—a sort of "inter-instructional space"—where, paradoxically, the very essence of programming resides. In any given situation, an instruction triggers an action whose consequences upon the preceding state are not directly discernible. The example provided by Jacques Arsac (2) illustrates this perfectly: consider the following sequence of three assignment statements—a sequence that is, in itself, quite simple:

```
X:=X+Y                          Y:=X-Y                          X:=X-Y
```

Signification is not directly accessible to us; we do not immediately perceive that this sequence exchanges the values of X and Y. Verifying the initial and final state of every instruction from one end of the program to the other constitutes "proving a program." Methodical programming thus ceases to be an empirical endeavor and becomes a scientific one. It applies Descartes' method to computing—the method that consists of breaking down a problem into smaller sub-problems until the solution appears self-evident. In so-called "structured programming," the program is thus partitioned into smaller problems to avoid having to tackle overly complex issues as a single monolithic block. The flowchart itself then becomes superfluous, as it diverts attention from the essential core by emphasizing a sequence of actions, whereas what truly takes precedence is the sequence of states.

Algorithmic Programming

Any "classic" program must facilitate the transformation of an initial state into a final state; this implies that the procedure to be followed must be perfectly determined—a requirement translated into sequences of instructions corresponding to successive steps. Once written, the program is utterly incapable of spontaneously generating a specific action not anticipated by the programmer. The primary task in writing a program, therefore, consists of defining an algorithm for the problem at hand—that is, a fully determined procedure capable of yielding a result across all possible scenarios. Finding the appropriate algorithm can be an extremely complex undertaking; indeed, the most frequent scenario—both in everyday life and in mathematics—is, unfortunately, precisely the one in which no known algorithm exists that is capable of solving the given problem. Alternatively, the known algorithm may be of such complexity that a computer is unable to execute it within a reasonable timeframe. For example, the algorithm enabling a computer to win every game of chess is known, yet it would require 10^120 calculations per game played (by way of comparison, the universe contains fewer than 10^78 atoms).

Artificial Intelligence

Once a viable algorithm has been discovered for a given problem, it intuitively seems to fall outside the realm of intelligence, insofar as a machine can "mindlessly" execute it. Conversely, when no such algorithm exists—or when the existing one is too complex—humans remain capable of making choices based on their experience and the specific nature of the situation at hand; a possible definition of intelligence, therefore, is to view it as the faculty to which humans turn in order to guide their choices.

The solution proposed by the field of artificial intelligence (AI) within computer science—in order to replicate such behaviors—involves incorporating a heuristic into the program; that is, a set of rules formulated by an expert in the relevant domain, designed to guide the computer toward a solution to the problem within a finite number of acceptable cases, by offering it a guiding principle that facilitates the elimination of unpromising possibilities. Instead of simply instructing it step-by-step on what to do, the system instills in it not only knowledge but also methodological procedures regarding the manipulation of that knowledge.

Whereas a classical computer program (i.e., purely algorithmic) integrates the problem-solving tools (the program code) and the relevant variables (the program data) into a single entity, this system distinguishes between: 1) the knowledge, and 2) the tools responsible for interpreting it. Expert Systems (ES) thus consist of a knowledge base that stores the expertise of the leading experts in a given field—knowledge formalized in the form of rules. The problem data is contained in a second component: the facts base. Based on these facts, the inference engine (an inference being a plausible line of reasoning) is tasked with applying and combining these rules to generate new facts; these new facts, in turn, trigger further rules, and so on, until a coherent answer is obtained. The essential task in AI, therefore, lies in formulating heuristic principles that are as general and powerful as possible—principles that are then complemented, in each specific instance, by particular rules.

THOUGHT PROCESSES AND ARTIFICIAL INTELLIGENCE

The Project of Artificial Intelligence

AI constitutes a project that is simultaneously philosophical, technical, and scientific. As previously mentioned, it presupposes a specific conception of intelligence. It sets forth a program of action: to construct an intelligence that conforms to this conception. It postulates that intelligence is a property inherent to certain automatic symbolic systems—a category within which it places both the human brain and the computer. As D. Andler (1) notes, this constitutes a much more precise—and therefore riskier, yet perhaps more fruitful—statement than the earlier proposition that the brain is a machine. Thus, AI is gradually coming to be described as the science capable of simulating, by means of machines, those human behaviors deemed intelligent. Its domains encompass perception, comprehension, decision-making, and learning. Ultimately, what is considered "intelligent reasoning" is reasoning that deliberates—that is, reasoning that judiciously selects inferences (for instance, in the form of production rules of the type: "If... <conditions>, Then... <conclusions>" and applies them to a body of knowledge. This knowledge may be defined (8) as symbolic representations of situations that are recognized, perceived, and identifiable.

The Failure of Behaviorism

Since AI seeks to reproduce behaviors, it draws resolutely from behaviorism, which—championing an ideal of objectivity—suggests that the subject matter of psychology consists solely of observable phenomena. However, as behaviorism proved difficult to generalize to complex human behaviors, the need for a substantial overhaul of behavioral psychology quickly became apparent. The stimulus-response (S-R) paradigm proved inadequate, as stimuli in this context are not merely interpreted by the system as physical interactions with the environment, but rather as an exchange of symbolic information. Furthermore, transforming symbolic information requires an internal space in which that information can be represented. Consequently, it was no longer sufficient to remain on the outside of Watson’s all-too-famous "black box."

Cognitive Psychology

The field of cognitive psychology intercedes between the stimulus and the response. It posits that internal representations are of just as great importance as environmental variations, and it aims to reconstruct them conceptually. The individual is viewed as an information-processing system that transforms information of a physical nature into information of a mental or representational nature. Information—a central concept—is to cognitive psychology what the stimulus is to behaviorism. Its theoretical frameworks are directly inspired by the functioning of computer systems, while—symmetrically—it influences certain lines of research in fundamental computer science. For instance, the heuristics employed in the General Problem Solver system developed by Newell and Simon are based, in part, on experimental research into problem-solving among human subjects. Compelled to address the issue of language comprehension, cognitive psychology pays close attention to linguistic research—particularly formal linguistic constructs (Miller, Chomsky)—and ultimately evolves into psycholinguistics. In its attempt to elucidate the architecture of the human mind, as well as the structure and laws governing knowledge representation, cognitive psychology draws a distinction between declarative knowledge and procedural knowledge. The former can be formalized as list structures—based on predicate logic—or in the form of scientific networks. The latter are amenable to formalization via a production system—that is, a set of condition-action rules. This approach focuses on the mechanisms by which elementary knowledge is transformed into integrated or compacted knowledge. It presupposes very high-level control processes and posits that the brain's capacity to take context into account constitutes the fundamental basis for its remarkable plasticity. These capabilities for utilizing contextual information arguably mark the precise, essential difference between human intelligence and the current state of AI.

The Epistemological Debate

G. Tiberghien’s article (8) sheds light on the complex epistemological landscape of AI. It strictly adheres to the experimental method and operates according to the model of the hypothetical-deductive sciences. But it carries within it a project that is both philosophical and technical, encroaching on the field of sciences concerned with the study of objects as well as on that of sciences concerned with the study of subjects. Moreover, can the machine or system to which we seek to attribute the equivalent of mental processes still be considered as belonging to the natural sciences, or must we transform our frames of reference and celebrate the birth, as H.A. Simon wished, of the sciences of the artificial, an emergence that would constitute a true epistemological break? If the cognitive sciences succeed in proving the existence of possible virtual knowledge, of "pure" knowledge, independent of the variations of the material systems that would be its origin, there would then no longer be a boundary between natural intelligence and artificial intelligence, but simply a difference in levels of complexity. It is precisely because cognitive science posits the existence of such virtual knowledge, which connects various natural biological, psychological, sociological, and economic systems, as well as the various artificial systems constructed by humans, that it finds unity and positions itself as the science of communication between different knowledge systems.

At the same time, the epistemological status of computer science is changing. Computer science struggled to establish itself as a science because, while its method, formal logic, was well-defined, its object remained ambiguous. By taking knowledge itself—virtual knowledge—as its object, rather than any particular logico-mathematical formalization of knowledge, it defines an object that is something other than multifaceted information.

THOUGHT AND MACHINE

Artificial intelligence has experienced extremely rapid development in recent years. However, J.-L. Le Moigne (5) notes that the three fundamental arguments for artificial intelligence had already been introduced by A. Turing: 1) the need to develop models of brain activity, a field in which the computational-symbolic approach, which seeks to understand the mind, is opposed to neuro-cybernetics, which seeks to understand the brain; 2) intelligence is considered the central concept of the mind-brain relationship; and 3) computing does not consist solely of arithmetically processing numbers but, more generally, of manipulating and processing symbols. The phrase "Intelligence as computation" condenses into two words the thesis that Newell and Simon would develop in 1975 during their famous Turing Conference.

We obviously remember the article that Turing himself published in October 1950 in Mind, a philosophical journal, "Computing Machinery and Intelligence," which has become a cornerstone of any anthology on AI. He began with this short sentence: "I propose we consider the question: can machines think?". We know that Turing proposed replacing this question with the game of imitation: if a digital computer is capable of simulating human behavior to such an extent that an interlocutor cannot determine who or what they are dealing with, then we may consider it to be endowed with human intelligence—a state equivalent to the capacity for thought. A series of significant anti-mechanist reactions sought to demonstrate that Turing's approach was unconvincing: performing the actions that humans perform does not necessarily entail the use of thought. One could not simply rely on the fact that a machine might perform just as well as a human being when substituted for one in a specific situation.

Among the principal critics, J.-R. Lucas (6) saw in Gödel's theorem the potential for a refutation of mechanism. If—according to this theorem—every sufficiently powerful logical system contains formulas that are "unprovable-within-the-system" (formulas that can be neither proven nor refuted from within that very system, unless the system itself is illogical)—a human intelligence can, nonetheless, recognize their truth. Consequently, no machine can serve as a complete model of the human mind, since the latter will always have the final say. Gödel's theorem thus represented "the Achilles' heel of cybernetic machines." Conversely, H. Putnam (7) believed that this line of argument amounted, quite simply, to a misapplication of Gödel's theorem. Turing took this criticism seriously and countered with the following argument: on the one hand, it has never been proven that the human mind is not itself fallible; and on the other, one can always construct a system in which a formula that is undecidable in the first system becomes decidable in the second. The human being's position relative to the machine is, therefore, no different from the machine's position relative to another machine. Lucas, in turn, refuted Turing’s argument by demonstrating that the challenge was not merely to triumph over a single, specific machine, but rather over any specific machine that anyone might choose to specify. And even if one were to equip the machine with a "Gödelizing operator"—that is, a system that internally incorporates the very procedure required to construct the Gödel formula for that system—man would still transcend this new machine, Gödelizing operator included...

Are the intellectual emotions evoked by self-referential systems linked to a perception of our own limitations, or do they stem from a formal convergence with the underlying structure of our own thought processes? Does the irreplaceable metaphorical value of Gödel’s theorem distance us from its truth-effects, or does it, conversely, draw us closer to them? It is upon this very thematic scale—among others—that Douglas Hofstadter (3) plays some twenty years later, in his book Gödel, Escher, Bach—originally published in 1979, though unfortunately translated into French only much later. One readily forgives the author’s occasional lyrical excesses, given the sheer infectiousness of his enthusiasm and jubilation.

The volume *Pensée et machine*—published by Champ Vallon in the "Milieu" series—contains a collection of essays selected by Alan Ross Anderson, with a specific focus on the philosophy of mind. Turing's seminal article, "Computing Machinery and Intelligence," appears therein in a French translation by P. Blanchard, under the title "Les ordinateurs et l'intelligence."

Hofstadter doesn't hesitate to bring together, around the central theme of self-referential systems, works as different as those of Escher, a graphic parable of Gödel's theorem, and Bach, allowing himself detours as unexpected as that of the double helix of DNA.

From the perspective of machine theory, there are two ways to approach the relationship between the machine and the brain, and they are a priori equivalent: the first starts with the machine and asks whether it can theoretically achieve the same capabilities as the mind. Conversely, the second takes psychic functioning as its starting point and asks whether or not it has the same operating characteristics as the machine. In reality, the two methods are very different, simply because, while we can perfectly determine how machines function, we ultimately know very little about psychic functioning. Turing's approach stemmed from a reflection on the capabilities of machines. He replaced the question of whether machines can think with that of whether they are capable of imitating human behavior. Douglas Hofstadter simultaneously explores the functioning of machines equipped with logical systems and the logic of thought processes. Far from remaining parallel, these two lines of inquiry intertwine, overlap, and mutually reinforce each other in a constant interplay.

Even if he does so metaphorically, Hofstadter does not hesitate to apply the characteristics of Gödel's theorem to psychic functioning. The far-reaching consequence of such a statement will be clear: it is psychic functioning that acquires the characteristics usually attributed to machines. As in logical systems, there is an impossibility, something irreducible leading to infinite repetition. But simultaneously, he argues that a system is intelligent when it finds within itself the capacity to modify itself and thus overcome its own contradictions. This is the very characteristic of what he calls a strange loop: infinite repetition and self-transcendence. And just when our minds thought they could finally rest on something tangible, a mysterious vertigo takes hold. Strange loops are a bit like Zen; sometimes we think we grasp what they are, at other times we feel we can never understand them. Is it any wonder, then, if, as D.H. suggests, they are at the very foundation of the structure of our thinking?

Like Zen, Douglas Hofstadter's book raises more questions than it answers. But let's be optimistic: if it has garnered attention in AI circles, sometimes to the point of being hailed as the first book of a new culture, it's because it suggests, by intuitively raising the paradoxical question of a boundary that is simultaneously insurmountable and yet surmountable (where it's difficult for us not to recognize the boundary between signifier and signified), that it suggests a possible breakthrough on the question of meaning, a question that has stumped cognitive science for the past fifteen years. In other words, it wouldn't be surprising if this book were the harbinger of the recognition of the place of the unconscious and that of the Subject, in the Lacanian sense, as well as the consideration of the psychoanalytic elaboration regarding language and the relationship between form and sense—particularly within its formalized productions. This would amount to elevating the unconscious to the very foundation of intelligent behavior. Yet, at this juncture, a question arises (one that we must dare to ask today): can computing ever attain sense? And, by way of a feedback effect, the question that computing poses to psychoanalysis: if it is true that the cause of the Subject lies in the signifier—that is to say, in arbitrary form—what is it that computing lacks to gain access to it? Or again: if the logico-mathematical formalizations of the formations of the unconscious are valid, what prevents their implementation on a computer capable—by that very token—of simulating psychic functioning? Does asserting that the computer cannot access meaning not *ipso facto* invalidate any logical formalization of the formations of the Unconscious? To date, no truly satisfactory answer has been provided to these inquiries—neither by computer scientists nor by psychoanalysts.

Setting aside the premature (though is it, in fact, merely a matter of time?) and certainly ill-posed question—"Can computers think?"—it now appears clear, in any case, that one of the major stakes of our immediate future lies in the expansion of research into the potential applications of computational methods within the human sciences.

BIBLIOGRAPHY

1. ANDLER D. — "Artificial Intelligence: A Second Wind." *Sciences et Avenir*. Special Issue (Hors-série), No. 53: 44–51.

2. ARSAC J. — "Program Design." *Pour la Science*, Nov. 1984, No. 85: 105–113.

3. HOFSTADTER D. — Gödel, Escher, Bach. *The Strands of an Eternal Garland*. Interédition, 1983, Paris.

4. KAY A. — "Software." *Pour la Science*, Nov. 1984, 85: 14–22.

5. LE MOIGNE J.L. — "The Genesis of Several New Sciences: From Artificial Intelligence to the Cognitive Sciences." In *Intelligence of Mechanisms, Mechanisms of Intelligence*, p. 229, Fayard, 1986, Paris.

6. LUCAS J-R. — "The Human Mind, the Machine, and Gödel." In *Thought and Machine*, Champ Vallon, 1983, Paris.

7. PUTNAM H. — "Thought and Machine." In *Thought and Machine*, Champ Vallon, 1983, Paris.

8. TIBERGHIEN G. — "Cognitive Psychology, Cognitive Science, and Knowledge Technology." In *Intelligence of Mechanisms, Mechanisms of Intelligence*, p. 173, Fayard, 1986, Paris.