ARTIFICIAL INTELLIGENCE:

SUBCOGNITION AS COMPUTATION*

by

Douglas R. Hofstadter
Computer Science Department
Indiana University
Bloomington, IN 47405

TECHNICAL REPORT NO. 132

ARTIFICIAL INTELLIGENCE:
SUBCOGNITION AS COMPUTATION

DOUGLAS R. HOFSTADTER
NOVEMBER, 1982

* Research reported herein was supported in part by the National Science Foundation under grant number MCS80-09201

ARTIFICIAL INTELLIGENCE: SUBCOGNITION AS COMPUTATION

by Douglas R. Hofstadter

Introduction

The philosopher John Searle has recently made quite a stir in the cognitive-science and philosophy-of-mind circles, with his celebrated "Chinese room" thought experiment, whose purpose is to reveal as illusory the aims of artificial intelligence (AI), and particularly to discredit what he labels "strong AI" -- the belief that a programmed computer can, in principle, be conscious. Various synonymous phrases could be substituted for "be conscious" here, such as: "think", "have a soul" (in a humanistic rather than a religious sense), "have an inner life", "have semantics" (as distinguished from "mere syntax"), "have content" (as distinguished from "mere form"), "have intentionality", "be something it is like something to be" (a somewhat ponderous yet appealing phrase due to philosopher T. Nagel), "have personhood", and others. Each of these phrases has its own peculiar set of connotations and imagery attached to it, as well as its own history and proponents. For our purposes, however, we shall consider them all as equivalent, and lump them all together, so that the claim of strong AI now becomes very strong indeed.

At the same time, various AI workers have been developing their own philosophies of what AI is, and have developed some useful terms and slogans to describe their endeavor. Some of them are: "information processing", "cognition as computation", "physical symbol system", "symbol manipulation", "expert system", and "knowledge engineering". There is some confusion as to what words like "symbol" and "cognition" actually mean, just as there is some confusion as to what words like "semantics" and "syntax" mean.

It is the purpose of this paper to try to delve into the meanings of such elusive terms, and at the same time to shed some light on the views of Searle, on the one hand, and Allen Newell and Herbert Simon, on the other hand -- visible AI pioneers who are responsible for several of the terms in the previous paragraph. The thoughts expressed herein were originally triggered by a paper called "Artificial Intelligence: Cognition as Computation" by Avron Barr. However, they can be read completely independently of that paper.

The questions are obviously not trivial, and certainly not resolvable in a single paper. Most of the ideas in this paper, in fact, were stated earlier and more fully in my book "Go:del, Escher, Bach: an Eternal Golden Braid". However, it seems worthwhile to extract a certain stream of ideas from that book and to enrich it with some more recent musings and examples, even if the underlying philosophy remains entirely the same. In order to do justice to these complex ideas, many topics must be interwoven, and they include the nature of symbols, meaning, thinking, perception, cognition, and so on. That explains why this paper is not three pages long.

Cognition versus Perception: The 100-millisecond Dividing Line

In Barr's original paper, AI is characterized repeatedly by the phrase "information-processing model of cognition". Although when I first heard that phrase years ago, I tended to accept it as defining the nature of AI, something has gradually come to bother me about it, and I would like to try to articulate that here.

Now what's in a word? What's to object to here? I won't attempt to say what's wrong with the phrase so much as try to show what I disagree with in the ideas of those who have promoted it; then, perhaps the phrase's connotations will float up to the surface so that other people can see why I am uneasy with it.

I think the disagreement can be put in its sharpest relief in the following way. In 1980, Simon delivered a lecture that I attended (the Procter Award Lecture for the Sigma Xi annual meeting in San Diego), and in it he declared (and I believe I am quoting him nearly verbatim): "Everything of interest in cognition happens above the 100-millisecond level -- the time it takes you to recognize your mother." Well, our disagreement is simple; namely, I take exactly the opposite viewpoint: "Everything interesting in cognition happens BELOW the 100-millisecond level -- the time it takes you to recognize your mother." To me, the major question of AI is this: "What in the world is going on to enable you to convert from 100,000,000 retinal dots into one single word "mother" in one tenth of a second?" Perception is where it's at!

The Problem of Letterforms: A Test Case for AI

The problem of intelligence, as I see it, is to understand the fluid nature of mental categories, to understand the invariant cores of percepts such as your mother's face, to understand the strangely flexible yet strong boundaries of concepts such as "chair" or the letter "a". Years ago, long before computers, Wittgenstein had already recognized the centrality of such questions, in his celebrated discussion of the nonpindownability of the meaning of the word "game". To emphasize this and make the point as starkly as I can, I hereby make the following claim:

The central problem of AI is the question: "What is the letter 'a'?"

Donald Knuth, on hearing me make this claim once, appended, "And what is the letter 'i'?" -- an amendment that I gladly accept. In fact, perhaps the best version would be this:

The central problem of AI is : "What are 'A' and 'I'?"

By making these claims, I am suggesting that, for any program to handle letterforms with the flexibility that human beings do, it would have to possess full-scale general intelligence.

Many people in AI might protest, pointing out that there already exist programs that have achieved expert-level performance in specialized domains without needing general intelligence. Why should letterforms be any different? My answer would be that specialized domains tend to obscure, rather than clarify, the distinction between strengths and weaknesses of a program. A familiar domain such as letterforms provides much more of an acid test.

It is strange that AI has said so little about this classic problem. To be sure, some work has been done. There are a few groups with interest in letters, but there has been no all-out effort to deal with this quintessential problem of pattern recognition. Since letterform understanding is currently the ultimate target of my own research project in AI, I would like to take a moment and explain why I see it as contrasting so highly with domains at the other end of the "expertise spectrum".

Each letter of the alphabet comes in literally thousands of different "official" versions (typefaces), not to mention millions, billions, trillions, of "unofficial" versions (those handwritten ones that you and I and everyone else produces all the time). There thus arises the obvious question, "How are all "a"'s like each other?" The goal of an AI project would be, of course, to give an exact answer in computational terms. However, even taking advantage of the vagueness of ordinary language, one is hard put to find a satisfactory intuitive answer, because we simply come up with phrases such as "They all have the same shape". Clearly, the whole problem is that they DON'T have the same shape. And it does not help to change "shape" to "form", or to tack on phrases such as "basically", "essentially", or "at a conceptual level".

There is also the less obvious question, "How are all the various letters in a single typeface related to each other?" This is a grand analogy problem if ever there were an analogy problem. One is asking for a "b" that is to the abstract notion of "b-ness" as a given "a" is to the abstract notion of "a"-ness. You have to take the qualities of a given "a" and, so to speak, "hold them loosely in the hand", as you see how they "slip" into variants of themselves as you try to carry them over to another letter. Here is the very hingepoint of thought, the place where one thing slips into alternate, subjunctive, variations on itself. Here, that "thing" is a very abstract concept -- namely, "the way that this particular shape manifests the abstract quality of being an 'a'". The problem of "a" is thus intimately connected with the problems of "b" through "z", and with that of stylistic consistency.

The existence of optical character readers might lead one to believe at first that the letter-recognition problem has been solved. If one considers the problem a little more carefully, however, one sees that the surface has barely been scratched. In truth, the way that most optical character recognition programs work is by a fancy kind of template matching, in which statistics are done to determine which character, out of a fixed repertoire of, say, 100 stored characters, is the "best match". This is about like assuming that the way I recognize my mother is by comparing the scene in front of me with stored memories of the appearances of tigers, cigarettes, hula hoops, gambling casinos, and can-openers (and of course all other things in the world simultaneously), and somehow instantly coming up with the "best match".

The Human Mind and Its Ability to Recognize and Reproduce Forms

The problem of recognizing letters of the alphabet is no less deep than that of recognizing your mother, even if it might seem so, given that the number of Platonic prototype items is on the small side (26, if one ignores all characters but the lowercase alphabet). One can even narrow it down further -- to just a handful. As a matter of fact, Godfried Toussaint, editor of the pattern recognition papers for the "IEEE Transactions", has said to me that he would like to put up a prize for the first program that could say correctly, of 20 characters that people easily can identify, which are "a"'s and which are "b"'s. To carry out such a task, a program cannot just recognize that a shape is an "a"; it has to see HOW that shape embodies "a"-ness. And then, as a test of whether the program really knows its letters, it would have carry "that style" over to the other letters of the alphabet. This is the goal of my research: to find out how to make letters slip in "similar ways to each other", so as to constitute a consistent artistic style in a typeface -- or simply a consistent way of writing the alphabet.

By contrast, most AI work on vision pertains to such things as aerial reconnaissance or robot guidance programs. This would suggest that the basic problem of vision is to figure out how to recognize textures and how to mediate between two and three dimensions. But what about the fact that although we are all marvelous face-recognizers, practically none of us can draw a face at all well -- even of someone we love? Most of us are flops at drawing even such simple things as pencils and hands and books. I personally have learned to recognize hundreds of Chinese characters (shapes that involve neither three dimensions nor textures) and yet, on trying to reproduce them from memory, find myself often drawing confused mixtures of characters, leaving out basic components, or worst of all, being unable to recall anything but the vaguest "feel" of the character and not being able to draw a single line.

Closer to home, most of us have read literally millions of, say, "k"'s with serifs, yet practically none of us can draw a "k" with serifs in the standard places. (This holds, of course, for any letter of the alphabet.) I suspect that many people -- perhaps most -- are not even consciously aware of the fact that there are two different types of lowercase "a" and of lowercase "g", just as many people seem to have a very hard time drawing a distinction between lowercase and uppercase letters, and a few have a hard time telling letters drawn forwards from letters drawn backwards.

How can such a fantastic "recognition machine" as our brain be so terrible at rendition? Clearly there must be something very complex going on, enabling us to ACCEPT things as members categories and to perceive HOW they are members of those categories, yet not enabling us to reproduce those things from memory. This is a deep mystery.

In his book "Pattern Recognition", M. Bongard concludes with a series of 100 puzzles for a visual pattern recognizer, whether human, machine, or alien, and to my mind it is no accident that he caps his set off with letterforms. In other words, he works his way up to letterforms as being at the pinnacle of visual recognition ability. There exists no pattern recognition program in the world today that can come anywhere close to doing those Bongard problems.

And yet, Barr cites Simon as writing the following statement:

> The evidence for that commonality [between the information processes that are employed by such disparate systems as computers and human nervous systems] is now overwhelming, and the remaining questions about the boundaries of cognitive science have more to do with whether there also exist nontrivial commonalities with information processing in genetic systems than with whether men and machines both think. Wherever the boundary is drawn, there exists today a science of intelligent systems that extends beyond the limits of any single species.

I find it difficult to understand how Simon can believe this, in an era when computers still cannot do basic kinds of "subcognitive" acts (acts that we feel are unconscious, acts that underlie cognition).

In another lecture in 1979 (the opening lecture of the first meeting of the Cognitive Science Society, also in San Diego), I recall Simon proclaiming that, despite much doubting by people not in the know, there is no longer any question as to whether computers can think. If he had meant that there should no longer be any question about whether machines may EVENTUALLY become able to think, or about whether we humans are machines (in some abstract sense of the term), then I would be in accord with his statement. But after hearing and reading such statements over and over again, I don't think that's what he meant at all. I get the impression that Simon genuinely believes that today's machines are intelligent, and that they really do think (or perform "acts of cognition" -- to use a bit of jargon that adds nothing to the meaning but makes it sound more scientific). I will come back to that shortly, since it is in essence the central bone of contention in this article, but first a few more remarks on AI domains.

Toy Domains, Technical Domains, Pure Science, and Engineering

There is a tendency in AI today towards flashy, splashy domains -- that is, towards developing programs that can do such things as medical diagnosis, geological consultation (for oil prospecting), designing of experiments in molecular biology, molecular spectroscopy, configuring of Vax installations, designing of VLSI circuits, and on and on. Yet there is no program that has common sense; no program that learns things that it has not been explicitly taught how to learn; no program that can recover gracefully from its own errors. The "artificial expertise" programs that do exist are rigid, brittle, inflexible. Like chess programs, they may serve a useful intellectual or even practical purpose, but despite much fanfare, they are not shedding much light on human intelligence. Mostly, they are being developed simply because various agencies or industries fund them.

This does not follow the traditional pattern of basic science. That pattern is to try to isolate a phenomenon, to reduce it to its simplest possible manifestation. For Newton, this meant the falling apple and the moon; for Einstein, the thought experiment of the trains and lightning flashes and, later, the falling elevator; for Mendel it meant the peas; and so on. You don't tackle the messiest problems before you've tackled the simpler ones; you don't try to run before you can walk. Or, to use a metaphor based on physics, you don't try to tackle a world with friction before you've got a solid understanding of the frictionless world.

Why do AI people eschew "toy domains"? Once, about ten years back, the MIT "blocks world" was a very fashionable domain. Roberts and Guzman and Waltz wrote programs that pulled 3-D blocks out of 2-D television screen dot matrices; Winston, building on their work, wrote a program that could recognize instantiations of certain concepts compounded from elementary blocks in that domain ("arch", "table", "house", and so on); Winograd wrote a program that could "converse" with a person about activities, plans, past events, and some structures in that circumscribed domain; Sussman wrote a program that could write and debug simple programs to carry out tasks in that domain, thus effecting a simple kind of learning. Why, then, did interest in this domain suddenly wane?

Surely no one could claim that the domain was exhausted. Every one of those programs exhibited glaring weaknesses and limitations and specializations. The domain was phenomenally far from being understood by a single, unified program. Here, then, was a nearly-ideal domain for exploring what cognition truly is -- and it was suddenly dropped. MIT was at one time doing truly basic research on intelligence, and then quit. Much basic research has been supplanted by large teams marketing what they vaunt as "knowledge engineering". Firmly grounded engineering is fine, but it seems to me that this type of engineering is not built upon the solid foundations of a science, but upon a number of recipes that have worked with some success in limited domains.

In my opinion, the proper choice of domain is the critical decision that an AI researcher makes, when beginning a project. If you choose to get involved in medical diagnosis at the expert level, then you are going to get mired down in a host of technical problems that have nothing to do with how the mind works. The same goes for the other earlier-cited ponderous domains that current work in expert systems involves. By contrast, if you are in control of your own domain, and can tailor it and prune it so that you keep the essence of the problem while getting rid of extraneous features, then you stand a chance of discovering something fundamental.

Early programs on the nature of analogy (Evans), sequence extrapolation (Simon and Kotovsky, among others), and so on, were moving in the right direction. But then, somehow, it became a common notion that these problems had been solved. Simply because Evans had made a program that could do some very restricted types of visual analogy problem "as well as a high school student", many people thought the book was closed. However, one need only look at Bongard's 100 to see how hopelessly far we are from dealing with analogies. One need only look at any collection of typefaces (look at any magazine's advertisements for a vast variety) to see how enormously far we are from understanding letterforms. As I claimed earlier, letterforms are probably the quintessential problem of pattern recognition. It is both baffling and disturbing to me to see so many people working on imitating cognitive functions at the highest level of sophistication when their programs cannot carry out cognitive functions at much lower levels of sophistication.

AI and the True Nature of Intelligence

There are some notable exceptions. The Schank group at Yale, whose original goal was to develop a program that could understand natural language, has been forced to "retreat", and to devote most of its attention to the organization of memory, which is certainly at the crux of cognition (because it is part of subcognition, incidentally) -- and the group has gracefully accommodated this shift of focus. I will not be at all surprised, however, if eventually the group is forced into yet further retreats -- in fact, all the way back to Bongard problems or the like. Why? Simply because their work (on such things as how to discover what "adage" accurately captures the "essence" of a story or episode) already has led them into the deep waters of abstraction, perception, and classification. These are the issues that Bongard problems illustrate so perfectly. Bongard problems are the idealized ("frictionless") versions of these critical questions.

It is interesting that Bongard problems are in actuality nothing other than a well-worked out set of typical IQ-test problems, the kind that Terman and Binet first invented 50 or more years ago. Over the years, many other less talented people have invented similar visual puzzles that had the unfortunate property of being filled with ambiguity and multiple answers. This (among other things) has given IQ tests a bad name. Whether or not IQ is a valid concept, however, there can be little question that the original insight of Terman and Binet -- that carefully constructed simple visual analogy problems probe close to the core mechanisms of intelligence -- is correct. Perhaps the political climate created a kind of knee-jerk reflex in many cognitive scientists to shy away from anything that smacked of IQ tests, since issues of cultural bias and racism began raising their ugly heads. But one need not be so Pavlovian as to jump whenever a visual analogy problem is placed in front of one. In any case, it will be good when AI people are finally driven back to looking at the insights of people working in the 20's, such as Wittgenstein and his "games", Koehler and Koffka and Wertheimer and their "gestalts", and Terman and Binet and their IQ-test problems.

I was saying that some AI groups seem to be less afraid of "toy domains", or more accurately put, they seem to be less afraid of stripping down their domain in successive steps, to isolate the core issues of intelligence that it involves. Aside from the Schank group, N. Sridharan at Rutgers has been doing some very interesting work on "prototype deformation" which, although it springs from work in legal reasoning in a quite messy real-world domain, has been abstracted into a form in which it is perhaps more like a toy domain (or, perhaps less pejorative-sounding, an "idealized domain") than at first would appear. The Lindsay-Norman-Rumelhart group at San Diego has been for years doing work on understanding errors, such as grammatical slips, typing errors, errors in everyday physical actions (such as winding your watch when you mean to be switching TV channels), for the insights it may offer into the underlying (subcognitive) mechanisms.

Then there are those people who are working on various programs for perception, whether visual or auditory. One of the most interesting was Hearsay II, a speech-understanding program developed at Carnegie-Mellon, Simon's home. It is therefore very surprising to me that that Simon, who surely was very aware of the wonderfully intricate and quite beautiful architecture of Hearsay II, could then make a comment indicating that perception and, in general, subcognitive (under 100 milliseconds) processes, "have no interest".

There are surely many other less publicized groups that are also working on more humble domains and on more pure problems of mind, but from looking at the proceedings of AI conferences one might get the impression that, indeed, computers must really be able to think these days, since after all, they are doing anything and everything cognitive -- from opthalmology to biology to chemistry to mathematics -- even discovering scientific laws from looking at tables of numerical data, to mention one project ("Bacon") that Simon has been involved in.

Expert Systems versus Human Fluidity

The problem is, AI programs are carrying out all these COGNITIVE activities in the absence of any SUBCOGNITIVE activity. There is no substrate that corresponds to what goes on in the brain. There is no fluid recognition and recall and reminding. These programs have no common sense, little sense of similarity or repetition or pattern. They can perceive some patterns as long as they have been anticipated -- and particularly, as long as the PLACE where they will occur has been anticipated -- but they cannot see patterns where nobody told them explicitly to look. They do not learn at a high level of abstraction.

This style is in complete contrast to how people are. People perceive patterns anywhere and everywhere, without knowing in advance where to look. People learn automatically in all aspects of life. These are just facets of common sense. Common sense is not an "area of expertise", but a general -- that is, domain-independent -- capacity that has to do with fluidity in representation of concepts, an ability to sift what is important from what is not, an ability to find unanticipated analogical similarities between totally different concepts ("reminding", as Schank calls it). We have a long way to go before our programs exhibit this cognitive style.

Recognition of one's mother's face is still nearly as much of a mystery as it was 30 years ago. And what about such things as recognizing family resemblances between people, recognizing a "French" face, recognizing kindness or earnestness or slyness or harshness in a face? Even recognizing age --

even sex! -- these are fantastically difficult problems! As Donald Knuth has pointed out, we have written programs that can do wonderfully well at what people have to work very hard at doing consciously (e.g., doing integrals, playing chess, medical diagnosis, and so on) -- but we have yet to write a program that remotely approaches our ability to do what we do WITHOUT thinking or training -- things like understanding a conversation partner with an accent at a loud cocktail party with music blaring in the background, while at the same time overhearing wisps of conversations in the far corner of the room. Or perhaps finding one's way through a forest on an overgrown trail. Or perhaps just doing some anagrams absent-mindedly while washing the dishes.

Asking for a program that can discover new scientific laws without having a program that can, say, do anagrams, is like wanting to go to the moon without having the ability to find your way around town. I do not make the comparison idly. The level of performance that Simon and his colleague Langley wish to achieve in Bacon is on the order of the greatest scientists. It seems they feel that they are but a step away from the mechanization of genius. After his Procter Lecture, Simon was asked by a member of the audience, "How many scientific lifetimes does a five-hour run of Bacon represent?" He replied, "Probably not more than one."

Anagrams and Epiphenomena

Well, I feel we're much further away from human-level performance than Simon does. I, for one, would like to see a program that does anagrams the way a person does. Why anagrams? Because they are a "toy domain" where some very significant subcognitive processes play the central role.

What I mean is this. When you look at a "Jumble" such as "telkin" in the newspaper, you immediately begin shifting around letters into tentative groups, making such stabs as "knitle", "klinte", "linket", "keltin", "tinkle" -- and then you notice that indeed, "tinkle" is a word. The part of this process that I am interested in is the part that precedes the recognition of "tinkle" as a word. It's that part that involves experimentation, based only on the "style" or "feel" of English words -- using intuitions about letter affinities, plausible clusters and their stabilities, syllable qualities, and so on. When you first read a jumble in the newspaper, you play around, rearranging, regrouping, reshuffling, in complex ways that you have no control over. In fact, it feels as if you throw the letters up into the air separately, and when they come down, they have somehow magically glommed together in some English-like word! It's a marvelous feeling -- and it is anything but cognitive, anything but conscious. (Yet, interestingly, YOU take credit for being good at anagrams, if you are good!)

It turns out that most literate people can handle jumbles (my term for single-word anagrams) of 5 or 6 letters, sometimes 7 or 8 letters. With practice, maybe even 10 or 12. But beyond that, it gets very hard to keep the letters in your head. It is especially hard if there are repeated letters, since one tends to get confused about which letters there are multiple copies of. (In one case, I rearranged the letters "dinnal" into "nadlid" -- incorrectly. You can try "raregarden", if you dare.) Now in one sense, the fact that the problem gets harder and harder with more and more letters is hardly surprising. It is obviously related to the famous "7 plus or minus 2" figure that psychologist George A. Miller first reported in connection with short-term memory capacity. But there are different ways of interpreting such a connection.

One way to think that this might come about is to assume that concepts for the individual letters get "activated" and then interact. When too many get activated simultaneously, then you get swamped with combinations and you drop some letters and make too many of others, and so on. This view would say that you simply encounter an explosion of connections, and your system gets overloaded. It does not postulate any explicit "storage location" in memory -- a fixed set of registers or data structures -- in which letters get placed and then shoved around. In this model, short-term memory and its associated "magic number" is an "epiphenomenon" (or "innocently emergent" phenomenon, as Daniel Dennett calls it), by which I mean it is a consequence that emerges out of the design of the system, a product of many interacting factors, something that was not necessarily known, predictable, or even anticipated to emerge at all. This is the view that I advocate.

A contrasting view might be to build a model of cognition in which you have an explicit structure called "short-term memory" which contains about seven (or five, or nine) "slots" into which certain data structures can be fitted, and when it is full, well, then, it is full and you have to wait until an empty slot opens up. This is one approach that has been followed by Newell and associates in work on production systems. The problem with this approach is that it takes something that clearly is a very complex consequence of underlying mechanisms, and simply plugs it in as an explicit structure, bypassing the question of what those underlying mechanisms might be. It is difficult for me to believe that any model of cognition based on such a "bypass" could be an accurate model.

When an operating system begins thrashing at around 35 users, do you tell the systems programmer, "Hey, go raise the thrashing-number in Tenex from 35 to 60, okay?"? The number 35 is not stored in some magic location in Tenex, so that it can be modified. That number comes out of a host of strategic decisions made by the designers of Tenex and the computer's hardware, and so on. There is no "thrashing-threshold dial" to crank on an operating system, unfortunately. Why should there be a "short-term-memory-size" dial on an intelligence? Why should 7 be a magic number built into the system explicitly from the start? If the size of short-term memory really were explicitly stored in our genes, then surely it would take only a simple mutation to reset the "dial" at 8 or 9 or 50, so that intelligence would evolve at ever-increasing rates. I doubt that AI people think that this is even remotely close to the truth; and yet they sometimes act as if it made sense to assume it is a close approximation to the truth.

It is standard practice for AI people to bypass epiphenomena ("collective phenomena", if you prefer) by simply installing structures that mimic the superficial features of those epiphenomena. (Such mimics are the "shadows" of genuine cognitive acts, as John Searle calls them in his provocative paper

"Minds, Brains, and Programs".) The expectation -- or at least the hope -- is for tremendous performance to issue forth; yet the systems lack the complex underpinning necessary.

The anagrams problem is one that exemplifies mechanisms of thought that AI people have not explored. How do those letters swirl among one another, fluidly and tentatively making and breaking alliances? Glomming together, then coming apart, almost like little biological objects in a cell. AI people have not paid much attention to such problems as anagrams. Perhaps they would say that the problem is "already solved". After all, a virtuoso programmer has made a program print out all possible words that anagrammize into other words in English. Or perhaps they would point out that in principle you can do an "alphabetize" followed by a "hash" and thereby retrieve, from any given set of letters, all the words they anagrammize into. Well, this is all fine and dandy, but it is really beside the point. It is merely a show of brute force, and has nothing to contribute to our understanding of how we actually do anagrams ourselves, just as most chess programs have absolutely nothing to say about how chess masters play (as de Groot, and later, Simon and coworkers have pointed out).

Is anagrams simply a trivial, silly, "toy" domain? Or is it serious? I maintain that it is a far purer, far more interesting domain than many of the complex real-world domains of the expert systems, precisely because it is so playful, so unconscious, so enjoyable, for people. It is obviously more related to creativity and spontaneity than it is to logical derivations, but that does not make it -- or the mode of thinking that it represents -- any less worthy of attention. In fact, because it epitomizes the unconscious mode of thought, I think it more worthy of attention.

In short, it seems to me that something fundamental is missing in the orthodox AI "information-processing" model of cognition, and that is some sort of substrate from which intelligence emerges as an epiphenomenon. Most AI people do not want to tackle that kind of underpinning work. Could it be that they really believe that machines already can think, already have concepts, already can do analogies? It seems that a large camp of AI people really do believe these things.

Not Cognition, but Subcognition, is Computational

Such beliefs arise, in my opinion, from a confusion of levels, exemplified by the title of Barr's paper: "Cognition as Computation". Am I really computing when I think? Admittedly, my neurons may be performing sums in an analog way, but does this pseudo-arithmetical hardware mean that the epiphenomena themselves are also doing arithmetic, or should be -- or even can be -- described in conventional computer-science terminology? Does the fact that taxis stop at red lights mean that traffic jams stop at red lights? One should not confuse the properties of objects with the properties of statistical ensembles of those objects. In this analogy, traffic jams play the role of thoughts, and taxis play the role of neurons or neuron-firings. It is not meant to be a serious analogy, only one that emphasizes that what you see at the top level need not have anything to do with the underlying swarm of activities bringing it into existence. In particular, something can be computational at one level, but not at another level.

Yet many AI people, despite considerable sophistication in thinking about a given system at different levels, still seem to miss this. Most AI work goes into efforts to build rational thought ("cognition") out of smaller rational thoughts (elementary steps of deduction, for instance, or elementary motions in a tree). It comes down to thinking that what we see at the top level of our minds -- our ability to think -- comes out of rational "information-processing" activity, with no deeper levels below that.

Many interesting ideas, in fact, have been inspired by this hope. I find much of the work in AI to be fascinating and provocative, yet somehow I feel dissatisfied with the overall trend. For instance, there are some people who believe that the ultimate solution to AI lies in getting better and better theorem-proving mechanisms in some predicate calculus. They have developed extremely efficient and novel ways of thinking about logic. Some people -- Simon and Newell particularly -- have argued that the ultimate solution lies in getting more and more efficient ways of searching a vast space of possibilities. (They refer to "selective heuristic search" as the key mechanism of intelligence.) Again, many interesting discoveries have come out of this.

Then there are others who think that the key to thought involves making some complex language in which pattern-matching or backtracking or inheritance or planning or reflective logic is easily carried out. Now admittedly, such systems, when developed, are good for solving a large class of problems, exemplified by such AI chestnuts as the missionary-and-cannibals problem, cryptarithmetic problems, retrograde chess problems, and many other specialized sorts of basically logical analysis. However, these kinds of techniques of building small logical components up to make large logical structures have not proven good for such things as recognizing your mother, or for drawing the alphabet in a novel and pleasing way.

One group of AI people who seem to have a different attitude consists of those who are working on problems of perception and recognition. There, the idea of coordinating many parallel processes is important, as is the idea that pieces of evidence can add up in a self-reinforcing way, so as to bring about the locking-in of a hypothesis that no one of the pieces of evidence could on its own justify. It is not easy to describe the flavor of this kind of program architecture without going into multiple technical details. However, it is very different in flavor from ones operating in a world where everything comes clean and precategorized -- where everything is specified in advance: "There are three missionaries and three cannibals and one boat and one river and..." which is immediately turned into a predicate-calculus statement or a frame representation, ready to be manipulated by an "inference engine". The missing link seems to be the one between perception and cognition, which I would rephrase as the link between subcognition and cognition, that gap between the sub-100-millisecond world and the super-100-millisecond world.

Earlier, I mentioned the brain and referred to the "neural substrate" of cognition. Although I am not pressing for a neurophysiological approach to AI, I am unlike many AI people in that I believe that any AI model eventually has to converge to brain-like hardware, or at least to an architecture that at some level of abstraction is "isomorphic" to brain architecture (also at some level of abstraction). This may sound empty, since that level could be anywhere, but I believe that the level at which the isomorphism must apply will turn out to be considerably lower than (I think) most AI people believe. This disagreement is intimately connected to the question of whether cognition should or should not be described as "computation".

Passive Symbols and Formal Rules

One way to explore this disagreement is to look at some of the ways that Simon and Newell express themselves about "symbols".

> At the root of intelligence are symbols, with their denotative power and their susceptibility to manipulation. And symbols can be manufactured of almost anything that can be arranged and patterned and combined. Intelligence is mind implemented by any patternable kind of matter. [Simon 1980, quoted by Barr, p. 17]

From this quotation and from others, one can see that to Simon and Newell, a symbol seems to be any token, any character inside a computer that has an ASCII code (a standard but arbitrarily assigned sequence of 7 bits). To me, by contrast, "symbol" connotes something with representational power. To them (if I am not mistaken), it would be fine to call a bit (inside a computer) or a neuron-firing a "symbol". However, I cannot feel comfortable with that usage of the term.

To me, the crux of the word "symbol" is its connection with the verb "to symbolize", which means "to denote", "to represent", "to stand for", and so on. Now in the quote above, Simon refers to the "denotative power" of symbols -- yet in another quote in his paper, Barr quotes Simon as saying (Newell & Simon 1976) that thought is "the manipulation of formal tokens". It is not clear to me which side of the fence they really are on.

It takes an immense amount of richness for something to represent something else. The letter "I" does not in and of itself stand for the person I am, or for the concept of selfhood. That quality comes to it from the way that the word behaves in the totality of the English language. It comes from a massively complex set of usages and patterns and regularities, ones that are regular enough for babies to be able to detect so that they too eventually come to say "I" to talk about themselves.

Formal tokens such as "I" or "HAMBURGER" are in themselves empty. They do not denote. Nor can they be made to denote in the full rich intuitive sense of the term by having them obey some rules. You can't simply push around some Pnames of Lisp atoms according to complex rules and hope to come out with genuine thought or understanding. (This, by the way, is probably a charitable way to interpret John Searle's point in his above-mentioned paper -- namely, as a rebellion against claims that programs that can manipulate tokens such as "John", "ate", "a", "hamburger" actually have understanding. Manipulation of empty tokens is not enough to create understanding -- although it is enough to imbue them with meaning in a LIMITED sense of the term, as I stress repeatedly in my book "Go:del, Escher, Bach" -- particularly in Chapters II through VI.)

Active Symbols and the Ant Colony Metaphor

So what is enough? What am I advocating? What do I mean by "symbol"? I gave an exposition of my concept of "active symbols" in Chapters XI and XII of "Go:del, Escher, Bach". However, the notion was first presented in the Dialogue "Prelude, Ant Fugue" in that book, which revolved about a hypothetical conscious ant colony. The purpose of the discussion was not to speculate about whether ant colonies are conscious or not, but to set up an extended metaphor for brain activity -- a framework in which to discuss the relationship between "holistic", or collective, phenomena, and the microscopic events that make them up.

One of the ideas that inspired the Dialogue has been stated by E. O. Wilson in his book "The Insect Societies" this way: "Mass communication is defined as the transfer, among groups, of information that a single individual could not pass to another." One has to imagine teams of ants cooperating on tasks, and that information passes from team to team that no ant is aware of (if ants indeed are "aware" of information at all -- but that is another question). One can carry this up a few levels, and imagine hyperhyperteams carrying and passing information that no hyperteam, not to mention team or solitary ant, ever dreamt of.

I feel it is critical to focus on collective phenomena, particularly on the idea that some information or knowledge or ideas can exist at the level of collective activities, while being totally absent at the lowest level. In fact, one can even go so far as to say that NO information exists at that lowest level. It is hardly an amazing revelation, when transported back to the brain: namely, that no ideas are flowing in those neurotransmitters that spark back and forth between neurons. Yet such a simple notion undermines the idea that thought and "symbol manipulation" are the same thing, if by "symbol" one means a formal token such as a bit or a letter or a Lisp Pname.

What is the difference? Why couldn't symbol manipulation -- in the sense that I believe Simon and Newell and many writers on AI mean it -- accomplish the same thing? The crux of the matter is that these people see symbols as lifeless, dead, passive objects -- things to be manipulated by some overlying program. I see symbols -- representational structures in the brain (or perhaps someday in a computer) -- as active, like the imaginary hyperhyperteams in the ant colony. THAT is the level at which denotation takes place, not at the level of the single ant. The single ant has no right to be called "symbolic", because its actions stand for nothing. (Of course, in a real ant colony, we have no reason to believe that teams at ANY level genuinely stand for objects outside the colony (or inside it, for that matter) -- but the ant colony metaphor is only a thinly disguised way of making discussion of the brain more vivid.)

Who Says Active Symbols are Computational Entities?

It is the vast collections of ants (read "neural firings", if you prefer) that add up to something genuinely symbolic. And who can say whether there exist rules -- formal, computational rules -- AT THE LEVEL OF THE TEAMS THEMSELVES (read "concepts", "ideas", "thoughts") that are of full predictive power in describing how they will flow? I am speaking of rules that allow you to ignore what is going on "down below", yet that still yield perfect or at least very accurate predictions of the teams' behavior.

To be sure, there are phenomenological observations that can be formalized to sound like rules that will describe, very vaguely, how those highest-level teams act. But what guarantee is there that we can skim off the full fluidity of the top-level activity of a brain, and encapsulate it -- without any lower substrate -- in the form of some computational rules?

To ask an analogous question, what guarantee is there that there are rules at the "cloud level" (more properly speaking, the level of cold fronts, isobars, trade winds, and so on) that will allow you to say accurately how the atmosphere is going to behave on a large scale? Perhaps there are no such rules; perhaps weather prediction is an intrinsically intractable problem. Perhaps the behavior of clouds is not expressible in terms that are computational AT THEIR OWN LEVEL, even if the behavior of the microscopic substrate -- the molecules -- is computational.

The premise of AI is that thoughts themselves are computational entities at their own level. At least this is the premise of the information-processing school of AI, and I have very serious doubts about it.

The difference between my active symbols ("teams") and passive "information-processing" symbols (ants, tokens) is that the active symbols flow and act on their own. In other words, there is no higher-level agent (read "program") that reaches down and shoves them around. Active symbols must incorporate within their own structures the wherewithal to trigger and cause actions. They cannot just be passive storehouses, bins, receptacles of data. Yet to Newell and Simon, it seems, even so tiny a thing as a bit is a symbol. This is brought out repeatedly in their writings on "physical symbol systems".

A good term for the little units that a computer manipulates (as well as for neuron firings) is "tokens". All computers are good at "token manipulation"; however, only some -- the appropriately programmed ones -- could support active symbols. (I prefer not to say that they would carry out "symbol manipulation", since that gets back to that image of a central program shoving around some passive representational structures.) The point is, in such a hypothetical program (and none exists as of yet) the symbols themselves are acting!

A simple analogy from ordinary programming might help to convey the level distinction that I am trying to make here. When a computer is running a Lisp program, does it do function calling? To say yes would be unconventional. The conventional intuition is that FUNCTIONS call other functions, and the computer is simply the hardware that SUPPORTS function-calling activity. In somewhat the same sense, although with much more parallelism, symbols activate, or trigger, or awaken, other symbols in a brain.

The brain itself does not "manipulate symbols"; the brain is the medium in which the symbols are floating, and in which they trigger each other. There is no central manipulator, no central program. There is simply a vast collection of "teams" -- patterns of neural firings that, like teams of ants, trigger other patterns of neural firings. The symbols are not "down there" at the level of the individual firings; they are "up here" where we do our verbalization. We feel those symbols churning within ourselves in somewhat the same way as we feel our stomach churning; we do not DO symbol manipulation by some sort of act of will, leave alone some set of logical rules of deduction. We cannot decide what we will next think of, nor how our thoughts will progress.

Not only are we not symbol manipulators; in fact, quite to the contrary, we are manipulated by our symbols! As Scott Kim has put it, rather than speak of "free will", perhaps it is more appropriate to speak of "free won't". This way of looking at things turns everything on its head, placing cognition -- that rational-seeming level of our minds -- where it belongs, namely as a consequence of much deeper processes of myriads of interacting subcognitive structures. The rational has had entirely too much made of it in AI research; it is time for some of the irrational and subcognitive to be recognized (no pun intended) for its pivotal role.

The Substrate of Active Symbols does Not Symbolize

"Cognition as computation" sounds right to me only if I interpret it quite liberally, namely, as meaning "cognition is an activity that can be supported by computational hardware". But if I interpret it more strictly as "cognition is an activity that can be achieved by a program that shunts around meaning-carrying objects called symbols in a complicated way", then I don't buy it. In my view, meaning-carrying objects won't submit to being shunted about (it's demeaning); meaning-carrying objects carry meaning only by virtue of being active, autonomous agents themselves. There can't be an overseer program, a pusher-around.

To paraphrase a question asked by neurophysiologist Roger Sperry, "Who shoves whom around inside the computer?" (He asked it of the cranium.) If some program shoves data structures around, then you can bet it's not carrying out cognition. Or more precisely, if the data structures are supposed to be MEANING-CARRYING, representational things, then it's not cognition. Of course, at SOME level of description, programs certainly will be shoving formal tokens around, but it's only agglomerations of such tokens EN MASSE that, above some unclear threshold of collectivity and cooperativity, achieve the status of genuine representation. At that stage, the computer is not shoving them around any more than our brain is shoving thoughts around! The thoughts themselves are causing flow. (This is, I believe, in agreement with Sperry's own way of looking at matters -- see, for instance, his article "Mind, Brain, and Humanist Values".) Parallelism and collectivity are of the essence, and in that sense, my response to the title of Barr's paper is no, cognition is NOT computation.

At this point, some people might think that I sound like John Searle, suggesting that there are elusive "causal powers of the brain" that cannot be captured computationally. I hasten to say that this is not my point of view at all! In my opinion, AI -- even Searle's "strong AI" -- is still possible, but thought will simply not turn out to be the formal dream of people inspired by predicate calculus or other formalisms. Thought is not a formal activity whose rules exist AT THAT LEVEL.

Many linguists have maintained that language is a human activity whose nature could be entirely explained at the linguistic level -- in terms of complex "grammars", without recourse or reference to anything such as thoughts or concepts. Now many AI people are making a similar mistake: they think that rational thought simply is composed of elementary steps, each of which has some interpretation as an "atom of rational thought", so to speak. That's just not what is going on, however, when neurons fire. On its own, a neuron firing has no meaning, no symbolic quality whatsoever. I believe that those elementary events at the bit level -- even at the Lisp function level (if AI is ever achieved in Lisp, something I seriously doubt) -- will have the same quality of HAVING NO INTERPRETATION. It is a level shift as drastic as that between molecules and gases that takes place when thought emerges from billions of in-themselves-meaningless neural firings.

A simple metaphor, hardly demonstrating my point but simply giving its flavor, is provided by Winograd's program SHRDLU, which, using the full power of a DEC-10, could deal with whole numbers up to ten in a conversation about the blocks world. It knew nothing -- at its "cognitive" level -- of larger numbers. Turing invents a similar example, a rather sly one, in his paper "Computing Machinery and Intelligence", where he has a human ask a computer to do a sum, and the computer pauses 30 seconds and then answers incorrectly. Now this need not be a ruse on the computer's part. It might genuinely have tried to add the two numbers at the SYMBOL LEVEL, and made a mistake, just as you or I might have, despite having neurons that can add fast.

The point is simply that the lower-level arithmetical processes out of which the higher level of any AI program are composed (the adds, the shifts, the multiplies, and so on) are completely shielded from its view. To be sure, Winograd could have artificially allowed his program to write little pieces of Lisp code that would execute and return answers to questions in English such as "What is 720 factorial?", but that would be similar to your trying to take advantage of the fact that you have billions of small analog adders in your brain, some time when you are trying to check a long grocery bill. You simply don't have access to those adders! You can't reach them.

Symbol Triggering Patterns are the Roots of Meaning

What's more, you OUGHTN'T to be able to reach them. The world is not sufficiently mathematical for that to be useful in survival. What good would it do a spear thrower to be able to calculate parabolic orbits when in reality there is wind and drag and the spear is not a point mass -- and so on? It's quite the contrary: a spear thrower does best by being able to imagine a cluster of approximations of what may happen, and anticipating some plausible consequences of them.

As Jacques Monod in "Chance and Necessity" and Richard Dawkins in "The Selfish Gene" both point out, the real power of brains is that they allow their owners to simulate a variety of plausible futures. This is to be distinguished from the EXACT prediction of eclipses by iterating differential equations step by step far into the future, with very high precision. The brain is a device that has evolved in a less exact world than the pristine one of orbiting planets, and there are always far more chances for the best-laid plans to "gang agley". Therefore, mathematical simulation has to be replaced by abstraction, which involves discarding the irrelevant, and making shrewd guesses based on analogy with past experience. Thus the symbols in a brain, rather than playing out a scenario precisely isomorphic to what actually will transpire, play out a few scenarios that are probable or plausible, or even some scenarios from the past that may have no obvious relevance other than as metaphors. (This brings us back to the "adages" of the Yale group.)

Once we abandon perfect mathematical isomorphism as our criterion for symbolizing, and suggest that symbol triggering-patterns are just as related to their suggestive value and their metaphorical richness, this severely complicates the question of what it means when we say that a symbol in the brain "symbolizes" anything. This is closely related to perhaps one of the subtlest issues, in my opinion, that AI should be able to shed light on, and that is the question "What is meaning?" This is actually the crucial issue that John Searle is concerned with in his earlier-mentioned attack on AI; although he camouflages it, and sometimes loses track of it by all sorts of evasive maneuvers, it turns out in the end (see his reply to Dennett in the June 24, 1982 New York Review) that what he is truly concerned with is the "fact" that "computers have no semantics" -- and he of course means "computers do not now have, and never will have, semantics". If he were talking only about the present, I would agree. However, he is making a point in principle, and I believe he is wrong there.

Where do the meanings of the so-called "active symbols", those giant "clouds" of neural activity in the brain, come from? To what do they owe their denotational power? Some people have maintained that it is because the brain is physically attached to sensors and effectors that connect it to the outside world, enabling those "clouds" to mirror the actual state of the world (or at least some parts of it) faithfully, and to affect the world outside as well, through the use of the body. I think that those things are PART of denotational power, but not its crux. When we daydream or imagine situations, when we dream or plan, we are NOT manipulating the concrete physical world, nor are we sensing it. In imagining fictional or hypothetical or even totally impossible situations we are still making use of, and contributing to, the meaningfulness of our symbolic neural machinery. However, the symbols do not symbolize specific, real, physical objects. The fundamental active symbols of the brain represent SEMANTIC CATEGORIES -- classes, in AI terminology.

Categories do not point to specific physical objects. However, they can be used as "masters" off of which copies -- instances -- can be rubbed, and then those copies are activated in various conjunctions; these activations then automatically trigger other instance-symbols into activations of various sorts (teams of ants triggering the creation of other teams of ants, sometimes themselves fizzling out). The overall activity will be semantic -- meaningful -- if it is isomorphic, not to some actual event in the real world, but to some event that is compatible with all the known constraints on the situation.

Those constraints are not at the molecular or any such fine-grained level; they are at the rather coarse-grained level of ordinary perception. They are to some extent verbalizable constraints. If I utter the Schankian cliche, "John went to a restaurant and ate a hamburger", there is genuine representational power in the patterns of activated symbols that your brain sets up, not because some guy named John actually went out and ate a hamburger (although, most likely, this is a situation that has at some time occurred in the world), but because the symbols, with their own "lives" (autonomous ways of triggering other symbols) will, if left alone, cause the playing-out of an imaginary yet realistic scenario. [Note added in press: I have it on good authority that one John Findling of Floyds Knobs, Indiana, did enter a Burger Queen restaurant and did eat one (1) hamburger. This fact, though helpful, would not, through its absence, have seriously marred the arguments of the present paper.]

Thus, the key thing that establishes meaningfulness is whether or not the semantic categories are "hooked up" in the proper ways so as to allow realistic scenarios to play themselves out on this "inner stage". That is, the triggering patterns of active symbols must mirror the general trends of how the world works as perceived on a macroscopic level, rather than mirroring the actual events that transpire.

Beyond Intuitive Physics: The Centrality of Slippability

Sometimes this capacity is referred to as "intuitive physics". Intuitive physics is certainly an important ingredient of the triggering patterns needed for an organism's comfortable survival. John McCarthy gives the example of someone able to avoid moving a coffee cup in a certain way, because they can anticipate how it might spill and coffee might get all over their clothes. Note that what is "computed" is a set of alternative rough descriptions for what might happen, rather than one exact "trajectory". This is the nature of intuitive physics.

However, as I stated earlier, there is much more required for symbols to have meaning than simply that their triggering patterns yield an intuitive physics. For instance, if you see someone in a big heavy leg cast and they tell you that their kneecap was acting up, you might think to yourself, "That's quite a nuisance, but it's nothing compared to my friend who has cancer." Now this connection is obviously caused by triggering patterns having to do with symbols representing health problems. But what does this have to do with the laws of motion governing objects or fluids? Precious little. Sideways connections like this, having nothing to do with causality, are equally much of the essence in allowing us to PLACE SITUATIONS IN PERSPECTIVE -- to compare what actually is with what, to our way of seeing things, "might have been" or might even come to be. This ability, no less than intuitive physics, is a central aspect of what meaning is.

This way that any situation that is perceived has of seeming to be surrounded by a cluster, a halo, of alternative versions of itself, of variations suggested by slipping any of a vast number of features that characterize the situation, seems to me to be at the dead center of thinking. Not much AI work seems to be going on at present (Schank's group excepted, perhaps -- and I ought to include myself as another maverick investigating these avenues) to mirror this kind of "slippability". This is an issue that I covered in some detail in "Go:del, Escher, Bach", under various headings such as "slippability", "subjunctive instant replays", "'almost' situations", "conceptual skeletons and conceptual mapping", "alternity" (a term due to George Steiner) and so on.

If we return to the metaphor of the ant colony, we can envision these "symbols with halos" as hyperhyperteams of ants, many of whose members are making what appear to be strange forays in random directions, like flickering tongues of flame spreading out in many directions at once. These tentative probes, which allow the possibility of all sorts of strange lateral connections as from "kneecap" to "cancer", have absolutely no detrimental effect on the total activity of the hyperhyperteam. In fact, quite to the contrary: the hyperhyperteam depends on its members to go wherever their noses lead them. The thing that saves the team -- what keeps it coherent -- is simply the regular patterns that are sure to emerge out of a random substrate when there are enough constituents. Statistics, in short.

Occasionally, some group of wandering scouts will cause a threshold amount of activity to occur in an unexpected place, and then a whole new area of activity springs up -- a new high-level team is activated (or, to return to the brain terminology, a new "symbol" is awakened). Thus, in a brain as in an ant colony, high-level activity spontaneously flows around, driven by the myriad lower-level components' autonomous actions.

AI's Goal Should be to Bridge the Gap between Cognition and Subcognition

Let me, for a final time, make clear how this is completely in contradistinction to standard computer programs. In a normal program, you can account for every single operation at the bit level, by looking "upwards" towards the top-level program. You can trace a high-level function call downwards: it calls subroutines that call other subroutines that call this particular machine-language routine that uses these words and in which this particular bit lies. So there is a high-level, global REASON why this particular bit is being manipulated.

By contrast, in an ant colony, a particular ant's foray is not the carrying-out of some global purpose. It has no interpretation in terms of the overall colony's goals; only when many such actions are considered at once does their statistical quality then emerge as purposeful, or interpretable. Ant actions are not the "translation into machine language" of some "colony-level program". No one ant is essential; even large numbers of ants are dispensable. All that matters is the statistics: thanks to it, the information moves around at a level far above that of the ants. Ditto for neural firings in brains. Not ditto for most current AI programs' architecture.

AI researchers started out thinking that they could reproduce all of cognition through a 100 percent top-down approach: functions calling subfunctions calling subsubfunctions, etc., until it all bottomed out in some primitives. Thus intelligence was thought to be hierarchically decomposable, with cognition at the top driving subcognition at the bottom. There were some successes and some difficulties -- difficulties particularly in the realm of perception. Then along came such things as production systems and pattern-directed inference. Here, some of bottom-up processing was allowed to occur within essentially still a top-down context. Gradually, the trend has been shifting. But there still is a large element of top-down quality in AI.

It is my belief that until AI has been stood on its head and is 100 percent bottom-up, it won't achieve the same level or type of intelligence as humans have. To be sure, when that kind of architecture exists, there will still be high-level, global, cognitive events -- but they will be epiphenomenal, like those in a brain. They will not in themselves be computational. Rather, they will be constituted out of, and driven by, many many smaller computational events, rather than the reverse. In other words, subcognition at the bottom will drive cognition at the top. And, perhaps most importantly, the activities that take place at that cognitive top level will neither have been written nor anticipated by any programmer.

Let me then close with a return to the comment of Simon's: "Nothing below 100 milliseconds is of interest in the study of cognition." I cannot imagine a remark about AI with which I could more vehemently disagree. Simon seems to be most concerned with having programs that can imitate chains of serial actions that come from verbal protocols of various experimental subjects. Perhaps, in some domains, even in some relatively complex and technical ones, people have come up with programs that can do this. But what about the simpler, noncognitive acts that in reality are the substrate for those cognitive acts? Whose program carries those out? At present, no one's. Why is this?

Because AI people have in general tended to cling to a notion that in some sense, thoughts obey formal rules at the thought level, just as George Boole believed that "the laws of thought" amounted to formal rules for manipulating propositions. I believe that this Boolean dream is at the root of the slogan "Cognition as computation" -- and I believe it will turn out to be revealed for what it is: an elegant chimera.

Acknowledgements

I would like to thank Daniel C. Dennett and Marsha J. Meredith for their careful reading of, and valuable comments on, earlier drafts of this paper.

References

Bongard, M. "Pattern Recognition". Hayden Book Co. (Spartan Books), 1970.

Boole, George. "The Laws of Thought". Dover.

Dawkins, R. "The Selfish Gene". Oxford Univ. Press, 1976.

Dennett, D. C. "Brainstorms". Bradford Books, 1978.

Evans, T. G. "A Program for the Solution of Geometric-Analogy Intelligence Test Questions". In M. Minsky, "Semantic Information Processing", MIT Press, 1968.

de Groot, Adriaan. "Thought and Choice in Chess". Mouton, 1965.

Hofstadter, D. R. "Go:del, Escher, Bach: an Eternal Golden Braid". Basic Books, 1979.

Hofstadter, D. R. "On Roles and Analogies in Human and Machine Thinking" ("Metamagical Themas"), Scientific American, Sept. 1981.

Hofstadter, D. R. "On Sphexishness, 'Jootsing', and Creativity" ("Metamagical Themas"), Scientific American, Sept. 1982.

Hofstadter, D. R. "On Variations on a Theme, Slippability, and Creativity" ("Metamagical Themas"), Scientific American, Oct. 1982.

Hofstadter, D. R. "The Tumult of Inner Voices", to be published by Southern Utah State College, 1982.

Hofstadter, D. R. "Who Shoves Whom Around Inside the Careenium?" Indiana University Computer Science Dept. Technical Report No. 130, July 1982. Also to appear in a forthcoming issue of "Synthese".

Hofstadter, D. R., G. A. Clossman, and M. J. Meredith. "Shakespeare's Plays Weren't Written by Him, But by Someone Else of the Same Name: An Essay on Intensionality and Frame-Based Knowledge Representation Systems." Indiana University Computer Science Dept. Technical Report No. 96, July, 1980.

Hofstadter, D. R. & D. C. Dennett. "The Mind's I". Basic Books, 1981.

Marais, Eugene. "The Soul of the White Ant". Penguin.

Meyer, Jean. "Essai d'application de certains mode'les cyberne'tiques a' la coordination chez les insectes sociaux". Insectes Sociaux, XIII, no. 2 (1966), p. 127.

Monod, J. "Chance and Necessity". Random House (Vintage), 1971.

Schank, Roger. "Dynamic Memory".

Searle, John R. "Minds, Brains, and Programs". (See Hofstadter & Dennett.)

Searle, John R. Response to Dennett in the letters section of the New York Review of Books, June 24, 1982.

Simon, H. A. 1980 Procter Lecture, published in The American Scientist, 1981.

Simon, H. A. and ??? Chase. In Simon and Chase (eds.), "Visual Information Processing".

Simon, H. A. and K. Kotovsky. "Human Acquisition of Concepts for Sequential Patterns". Psychological Review, Vol. 70, No. 6 (1963), pp. 534-545.

Sloman, Aaron. "The Computer Revolution in Philosophy". Harvester, 1978.

Sperry, Roger. "Mind, Brain, and Humanist Values", in John R. Platt (ed.) "New Views on the Nature of Man", University of Chicago Press, 1965.

Steiner, George. "After Babel". Oxford Univ. Press, 1975.

Sussman, G. "A Computer Model of Skill Acquisition". American Elsevier, 1975.

Turing, Alan. "Computing Machinery and Intelligence". Mind, Vol. LIX (1950), p. 236.

Wheeler, William Morton. "The Ant Colony as an Organism". Journal of Morphology 22, 2 (1911), pp. 307-325.

Wilson, E. O. "The Insect Societies". Harvard University Press, 1971.

Winograd, T. "Understanding Natural Language". Academic Press, 1972.

Winston, P. (ed.) "The Psychology of Computer Vision". McGraw-Hill, 1975.