Volume 19 Number 4 November 1995 ISSN 0350-5596

Informatica

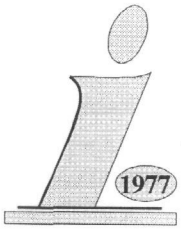
An International Journal of Computing and Informatics

Special Issue:
Mind <> Computer

Were Dreyfus and Winograd right?

Guest Editors:

Matjaž Gams (Europe, Africa)
Marcin Paprzycki (Americas)
Xindong Wu (Asia, Australia)



The Slovene Society Informatika, Ljubljana, Slovenia

Informatica 19 (1995) Number 4, pp. 423–665

Informatica

An International Journal of Computing and Informatics

Basic info about Informatica and back issues may be FTP'from ftp.arnes.si in magazines/informatica ID: anonymous PASSWORD: <your mail address>
FTP archive may be also accessed with WWW (worldwide web) clients with URL: `http://www2.ijs.si/~mezi/informatica.html`

Subscription Informatio Informatica (ISSN 0350-5596) is published four times a year in Spring, Summer, Autumn, and Winter (4 issues per year) by the Slovene Society Informatika, Vožarski pot 12, 61000 Ljubljana, Slovenia.
The subscription rate for 1995 (Volume 19) is
– DEM 50 (US$ 35) for institutions,
– DEM 25 (US$ 17) for individuals, and
– DEM 10 (US$ 7) for students
plus the mail charge DEM 10 (US$ 7).
Claims for missing issues will be honored free of charge within six months after the publication date of the issue.

LaTeX Tech. Support: Borut Žnidar, DALCOM d.o.o., Stegne 27, 61000 Ljubljana, Slovenia.
Lectorship: Fergus F. Smith, AMIDAS d.o.o., Cankarjevo nabrežje 11, Ljubljana, Slovenia.
Printed by Biro M, d.o.o., Žibertova 1, 61000 Ljubljana, Slovenia.

Orders for subscription may be placed by telephone or fax using any major credit card. Please call Mr. R. Murn, Jožef Stefan Institute: Tel (+386) 61 1773 900, Fax (+386) 61 219 385, or use the bank account number 900–27620–5159/4 Ljubljanska banka d.d. Slovenia (LB 50101-678-51841 for domestic subscribers only).

According to the opinion of the Ministry for Informing (number 23/216-92 of March 27, 1992), the scientific journal Informatica is a product of informative matter (point 13 of the tariff number 3), for which the tax of traffic amounts to 5%.

Informatica is published in cooperation with the following societies (and contact persons):
Robotics Society of Slovenia (Jadran Lenarčič)
Slovene Society for Pattern Recognition (Franjo Pernuš)
Slovenian Artificial Intelligence Society (Matjaž Gams)
Slovenian Society of Mathematicians, Physicists and Astronomers (Bojan Mohar)
Automatic Control Society of Slovenia (Borut Zupančič)
Slovenian Association of Technical and Natural Sciences (Janez Peklenik)

Informatica is surveyed by: AI and Robotic Abstracts, AI References, ACM Computing Surveys, Applied Science & Techn. Index, COMPENDEX*PLUS, Computer ASAP, Cur. Cont. & Comp. & Math. Sear., Engineering Index, INSPEC, Mathematical Reviews, Sociological Abstracts, Uncover, Zentralblatt für Mathematik und ihre Grenzgebiete., Linguistics and Language Behaviour Abstracts, Cybernetica Newsletter

The issuing of the Informatica journal is financially supported by the Ministry for Science and Technology, Slovenska 50, 61000 Ljubljana, Slovenia.

MIND <> COMPUTER: INTRODUCTION TO THE SPECIAL ISSUE

Matjaž Gams, Marcin Paprzycki, Xindong Wu

see FTP: ftp.arnes.si magazines/informatica anonymous your-mail
or WWW: http://www2.ijs.si/~mezi/informatica.html

This special issue of *Informatica* on *Mind <> Machine* aims to reevaluate the soundness of current AI research, especially the heavily disputed strong-AI paradigm, and to pursue new directions towards achieving true intelligence. It is a brainstorming issue about core ideas that will shape future AI. We have tried to include critical papers representing different positions on these issues.

Submissions were invited in all subareas and on all aspects of AI research and its new directions, especially:

- the current state, positions, and true advances achieved in the last 5-10 years in various subfields of AI (as opposed to parametric improvements),
- the trends, perspectives and foundations of artificial and natural intelligence, and
- strong AI vs. weak AI and the reality of most current "typical" publications in AI.

Papers accepted for the special issue include invited papers from Agre, Dreyfus, Gams, Michie, Winograd and Wu, and regular submissions. The invited papers were refereed in the same way as regular submissions, and all authors were asked to accommodate comments from referees. The accepted papers are grouped into the following three categories.

A. Overview and General Issues

Making a Mind vs. Modelling the Brain: AI Back at a Branchpoint by H.L. Dreyfus and S.E. Dreyfus, and *Thinking machines: Can there be? Are we?* by T. Winograd, are both unique and worth reading again and again. Indeed, they present the motto of this special issue – were not H.L. Dreyfus, S.E. Dreyfus and T. Winograd right about this issue years ago? Were the attacks on them by the strong-AI community and large parts of the formal-sciences community unjustified? We believe the answer is yes.

"Strong AI": An Adolescent Disorder by D. Michie advocates an integrative approach – let us forget about differences and keep doing interesting things.

Artificial Selfhood: The Path to True Artificial Intelligence by B. Goertzel rejects formal logic and advocates designing complex self-aware systems.

Strong vs. Weak AI by M. Gams presents an overview of the antagonistic approaches and proposes an AI version of the Heisenberg principle delimiting strong from weak AI.

A Brief Naive Psychology Manifesto by S. Watt argues for naive commonsense psychology, by analogy to naive physics. People understand physics and psychology even in their childhood without any formal logic or equations.

Stuffing Mind into Computer: Knowledge and Learning for Intelligent Systems by K.J. Cherkauer analyses knowledge acquisition and learning as the key issues necessary for designing intelligent computers.

Has Turing Slain the Jabberwock? by L. Marinoff attacks strong AI through slaying Turing and Jabberwock.

The papers in this section are a mixture of interdisciplinary approaches, from computer- to cognitive sciences. The average paper takes a critical stand against strong AI. However, the level of criticism and acclaim for intelligent digital computers varies.

B. New Approaches

Computation and Embodied Agency by P.E. Agre analyses computational theories of agents' interactions with their environments.

Methodological Considerations on Modeling Cognition and Designing Human-Computer Interfa-

ces - An Investigation from the Perspective of Philosophy of Science and Epistemology by M.F. Peschl investigates the role of representation in both cognitive modeling and the development of human-computer interfaces.

Knowledge Objects by X. Wu, S. Ramakrishnan and H. Schmidt introduces knowledge objects as a step further from programming objects.

Modeling Affect: The Next Step in Intelligent Computer Evolution by S. Walczak advocates implementing features such as affects in order to design intelligent programming systems.

The Extracellular Containment of Natural Intelligence: A New Direction for Strong AI by R.L. Amoroso is one of the rare papers closely connecting physics and AI in this issue.

Quantum Intelligence, QI; Quantum Mind, QM by B. Souček presents and defines concepts of quantum intelligence and quantum mind.

Representations, Explanations, and PDP: Is Representation-Talk Really Necessary? by R.S. Stufflebeam addresses the connectionist approach. What has happened to the neural-network wave of optimism?

C. Computability and Form vs. Meaning

Is Consciousness a Computational Property? by G. Caplain proposes a detailed argument to show that mind can not be computationally modeled.

Cracks in the Computational Foundations by P. Schweizer claims that computational procedures are not constitutive of the mind, and thus cannot play a fundamental role in AI.

Gödel's Theorems for Minds and Computers by D. Bojadžiev, presents an overview of the uses of Gödel's theorems, claiming that they apply equally to humans and computers.

On the Computational Model of the Mind by M. Radovan examines various strengths and shortcomings of computers and minds. Although computers in many ways exceed natural mind, brains still have quite a few aces left.

What Internal Languages Can't Do by P. Hipwell analyses the limitations of internal representation languages in contrast with the brain's representations.

Consciousness and Understanding in the Chinese Room by S. Gozzano proposes yet another reason why Searle's Chinese rooms present a hypothetical situation only.

Acknowledgements

The following reviewers are gratefully thanked for their time and effort to make this special issue a reality:

- Kenneth Aizawa
- Alan Aliu
- Balaji Bharadwaj
- Leslie Burkholder
- Frada Burstein
- Sait Dogru
- Mark Druzdzel
- Stavros Kokkotos
- Kevin Korb
- Timothy Menzies
- Madhav Moganti
- John Mueller
- Hari Narayanan
- James Pomykalski
- David Robertson
- Olivier de Vel
- John Weckert
- Stefan Wrobel

Making a Mind vs. Modeling the Brain: AI Back at a Branchpoint

Hubert L. Dreyfus and Stuart E. Dreyfus
University of California,
Berkeley

Keywords: mind, brain, AI directions

Edited by: Matjaž Gams
Received: October 17, 1994 **Revised:** October 4, 1995 **Accepted:** October 18, 1995

Nothing seems more possible to me than that people some day will come to the definite opinion that there is no copy in the... nervous system which corresponds to a *particular* thought, or à *particular* idea, or memory.[1]

Information is not stored anywhere in particular. Rather it is stored everywhere. Information is better thought of as "evoked" than "found".[2]

In the early 1950s, as calculating machines were coming into their own, a few pioneer thinkers began to realize that digital computers could be more than number crunchers.[3] At that point two opposed visions of what computers could be, each with its correlated research program, emerged and struggled for recognition. One faction saw computers as a system for manipulating mental symbols; the other, as a medium for modeling the brain. One sought to use computers to instantiate a formal representation of the world; the other, to simulate the interactions of neurons. One took problem solving as its paradigm of intelligence; the other, learning. One utilized logic, the other statistics. One school was the heir to the rationalist, reductionist tradition in philosophy; the other viewed itself as idealized, holistic neuro-science.

The rallying cry of the first group was that both minds and digital computers were physical symbol systems. By 1955 Allen Newell and Herbert Simon, working at the RAND Corporation, had concluded that strings of bits manipulated by a digital computer could stand for anything – numbers, of course, but also features of the real world. Moreover, programs could be used as rules to represent relations between these symbols, so that the system could infer further facts about the represented objects and their relations. As Newell put it recently in his account of the history of issues in AI:

The digital-computer field defined computers as machines that manipulated numbers. The great thing was, adherents said, that everything could be encoded into numbers, even instructions. In contrast, the scientists in AI saw computers as machines that manipulated symbols. The great thing was, they said, that everything could be encoded into symbols, even numbers.[4]

This way of looking at computers became the basis of a way of looking at minds. Newell and Simon hypothesized that the human brain and the digital computer, while totally different in structure and mechanism, had, at the appropriate level of abstraction, a common functional description. At this level, both the human brain and the appropriately programmed digital computer could be seen as two different instantiations of a single species of device – one which generated

[1] L. Wittgenstein, *Last Writings on the Philosophy of Psychology, Vol. I*, Chicago University Press, 1982, #504, p. 66e. (Translation corrected).

[2] Rumelhart and Norman, "A Comparison of Models," *Parallel Models of Associative Memory*, Hinton and Anderson eds., Lawrence Erlbaum Associates, Publishers, 1981, p. 3.

[3] First published as Dreyfus, H. L. & Dreyfus, S. E. (1988), Making a mind versus modelling the brain: Artificial intelligence back at a branchpoint, *Daedalus*, 117(1):185–197. Reprinted with permission.

[4] Allen Newell, "Intellectual Issues in the History of Artificial Intelligence", in *The Study of Information: Interdisciplinary Messages*, F. Machlup and U. Mansfield, eds. (New York: John Wiley and Sons, 1983), p. 196.

intelligent behavior by manipulating symbols by means of formal rules. Newell and Simon stated their view as an hypothesis:

The Physical Symbol System Hypothesis. A physical symbol system has the necessary and sufficient means for general intelligent action.

By "necessary" we mean that any system that exhibits general intelligence will prove upon analysis to be a physical symbol system. By "sufficient" we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence.[5]

Newell and Simon trace the roots of their hypothesis back to Frege, Russell, and Whitehead[6] but, of course, Frege and company were themselves heirs to a long atomistic, rationalist tradition. Descartes already assumed that all understanding consisted in forming and manipulating appropriate representations, that these representations could be analyzed into primitive elements (*naturas simplices*), and that all phenomena could be understood as a complex combinations of these simple elements. Moreover, at the same time, Hobbes implicitly assumed that the elements were formal elements related by purely syntactic operations, so that reasoning could be reduced to calculation. "When a man *reasons*, he does nothing else but conceive a sum total from addition of parcels," Hobbes wrote, "for REASON... is nothing but reckoning..."[7] Finally Leibniz, working out the classical idea of mathesis – the formalization of everything –, sought support to develop a universal symbol system, so that "we can assign to every object its determined characteristic number".[8] According to Leibniz, in understanding we analyze concepts into more simple elements. In order to avoid a regress of simpler and simpler elements, there must be ultimate simples in terms of which all complex concepts can be understood. Moreover, if concepts are to apply to the world, there must be simple features which these elements represent. Leibniz envisaged "a kind of alphabet of human thoughts"[9] whose "characters must show, when they are used in demonstrations, some kind of connection, grouping and order which are also found in the objects."[10]

Ludwig Wittgenstein, drawing on Frege and Russell, stated the pure form of this syntactic, representational view of the relation of the mind to reality in his *Tractatus Logico-Philosophicus*. He defined the world as the totality of logically independent atomic facts:

1.1. The world is the totality of facts, not of things.

Facts, in turn, were exhaustively analyzable into primitive objects.

2.01. An atomic fact is a combination of objects...

2.0124. If all objects are given, then *thereby* all atomic facts are given.

These facts, their constituents, and their logical relations were represented in the mind.

2.1. We make to ourselves pictures of facts.

2.15. That the elements of the picture are combined with one another in a definite way,. represents that the things are so combined with one another.[11]

AI can be thought of as the attempt to find the primitive elements and logical relations in the subject (man or computer) which mirror the primitive objects and their relations which make up the world. Newell and Simon's physical symbol system hypothesis in effect turns the Wittgensteinian vision – which is itself the culmination of the classical rationalist philosophical tradition – into an empirical claim, and bases a research program on it.

The opposed intuition, that we should set about creating artificial intelligence by modeling the brain not the mind's symbolic representation of the world, drew its inspiration not from philosophy but from what was soon to be called neuroscience. It was directly inspired by the work of D.O. Hebb who in 1949 suggested that a mass of neurons could learn if, when neuron A and neuron B were simultaneously excited, that increased the strength of the connection between them.

This lead was followed by Frank Rosenblatt

[5]Allen Newell and Herbert Simon, "Computer Science as Empirical Inquiry: Symbols and Search", reprinted in *Mind Design*, John Haugeland, ed., (Cambridge: Bradford/MIT Press, 1981), p. 41.

[6]Ibid., p. 42.

[7]Hobbes, *Leviathan*, (New York: Library of Liberal Arts, 1958), p. 45.

[8]Leibniz, *Selections*, ed. Philip Wiener (New York: Scribner, 1951), p. 18.

[9]Ibid., p. 20.

[10]Ibid., p. 10.

[11]L. Wittgenstein, *Tractatus Logico-Philosophicus*, (London: Routledge and Kegan Paul, 1960).

who reasoned that since intelligent behavior based on our representation of the world was likely to be hard to formalize, AI should rather attempt to automate the procedures by which a network of neurons learns to discriminate patterns and respond appropriately. As Rosenblatt put it:

The implicit assumption of the symbol manipulating research program is that it is relatively easy to specify the behavior that we want the system to perform, and that the challenge is then to design a device or mechanism which will effectively carry out this behavior... It is both easier and more profitable to axiomatize the *physical system* and then investigate this system analytically to determine its behavior, than to axiomatize the *behavior* and then design a physical system by techniques of logical synthesis.[12]

Another way to put the difference between the two research programs is that those seeking symbolic representations were looking for a formal structure that would give the computer the ability to solve a certain class of problems or discriminate certain types of patterns. Rosenblatt, on the other hand, wanted to build a physical device, or to simulate such a device on a digital computer, which then could generate its own abilities.

Many of the models which we have heard discussed are concerned with the question of what logical structure a system must have if it is to exhibit some property, X. This is essentially a question about a static system...

An alternative way of looking at the question is: what kind of a system can *evolve* property X ? I think we can show in a number of interesting cases that the second question can be solved without having an answer to the first.[13]

Both approaches met with immediate and startling success. Newell and Simon succeeded by 1956 in programming a computer using symbolic representations to solve simple puzzles and prove theorems in the propositional calculus. On the basis of these early impressive results it looked like the physical symbol system hypothesis was about to be confirmed, and Newell and Simon were understandably euphoric. Simon announced:

It is not my aim to surprise or shock you... But the simplest way I can summarize is to say that there are now in the world machines that think, that learn and that create. Moreover, their ability to do these things is going to increase rapidly until – in a visible future – the range of problems they can handle will be coextensive with the range to which the human mind has been applied.[14]

He and Newell explained:

We now have the elements of a theory of heuristic (as contrasted with algorithmic) problem solving; and we can use this theory both to understand human heuristic processes and to simulate such processes with digital computers. Intuition, insight, and learning are no longer exclusive possessions of humans: any large high-speed computer can be programmed to exhibit them also.[15] Heuristic rules are rules that when used by human beings are said to be based on experience or judgment. Such rules frequently lead to plausible solutions to problems or increase the efficiency of a problem-solving procedure. Whereas algorithms guarantee a correct solution (if there is one) in a finite time, heuristics only increase the likelihood of finding a plausible solution.

Rosenblatt put his ideas to work in a type of device which he called a perceptron.[16] By 1956

[12]Frank Rosenblatt, "Strategic Approaches to the Study of Brain Models," *Principles of Self-Organization*, H. von Foerster, ed., (Pergamon Press, 1962), p. 386.

[13]Ibid., p. 387.

[14]Herbert Simon and Allen Newell, "Heuristic Problem Solving: The Next Advance in Operations Research", *Operations Research*, Vol. 6 (January- February 1958), p. 6.

[15]Ibid.

[16]David Rumelhart and James McClelland in their recent book, *Parallel Distributed Processing*, describe the perceptron as follows: "Such machines consist of what is generally called a *retina*, an array of binary inputs sometimes taken to be arranged in a two-dimensional spatial layout; a set of *predicates*, a set of binary threshold units with fixed connections to a subset of units in the retina such that each predicate computes some local function over the subset of units to which it is connected; and one or more decision units, with modifiable connections to the predicates." (p. 111).

They contrast the way a parallel distributed processing (PDP) model like the perceptron stores information with the way information is stored by symbolic representation. "In most models, knowledge is stored as a static copy of a pattern. Retrieval amounts to finding the pattern in long-term memory and copying it into a buffer or working memory. There is no real difference between the stored representation in long-term memory and the active representation in working memory. In PDP models, though, this is not the case. In these models, the patterns themselves are not stored. Rather, what is stored is the *connection strengths* between units that allow these patterns to be re-created." (p. 31) "Knowledge about any individual pattern is not stored in the connections of a special unit

Rosenblatt was able to train a perceptron to classify certain types of patterns as similar and to separate these from other patterns which were dissimilar. By 1959 he too was jubilant and felt his approach had been vindicated:

It seems clear that the... perceptron introduces a new kind of information processing automaton: For the first time, we have a machine which is capable of having original ideas. As an analogue of the biological brain, the perceptron, more precisely, the theory of statistical separability, seems to come closer to meeting the requirements of a functional explanation of the nervous system than any system previously proposed... As concept, it would seem that the perceptron has established, beyond doubt, the feasibility and principle of nonhuman systems which may embody human cognitive functions... The future of information processing devices which operate on statistical, rather than logical, principles seems to be clearly indicated.[17]

In the early sixties both approaches looked equally promising, and both made themselves equally vulnerable by making exaggerated claims. Yet the result of the internal war between the two research programs was surprisingly asymmetrical. By 1970 the brain simulation research which had its paradigm in the perceptron was reduced to a few, lonely, underfunded efforts, while those who proposed using digital computers as symbol manipulators had undisputed control of the resources, graduate programs, journals, symposia, etc. that constitute a flourishing research program.

Reconstructing how this came about is complicated by the myth of manifest destiny an on-going research program generates. Thus it looks to the victors as if symbolic information processing won out because it was on the right track, while the neural net approach lost because it simply didn't work. But this account of the history of the field is a retroactive illusion. Both research programs had ideas worth exploring and both had deep, unrecognized problems.

Each position had its detractors and what they said was essentially the same: each approach had shown that it could solve certain easy problems but that there was no reason to think that either group could extrapolate its methods to real world complexity. Indeed, there was evidence that as problems got more complex the computation required by both approaches would grow exponentially and so soon become intractable. Marvin Minsky and Seymour Papert said in 1969 of Rosenblatt's perceptron:

Rosenblatt's schemes quickly took root, and soon there were perhaps as many as a hundred groups, large and small, experimenting with the model...

The results of these hundreds of projects and experiments were generally disappointing, and the explanations inconclusive. The machines usually work quite well on very simple problems but deteriorate very rapidly as the tasks assigned to them get harder.[18]

Three years later, Sir James Lighthill, after reviewing work using heuristic programs such as Simon's and Minsky's reached a strikingly similar negative conclusion:

Most workers in AI research and in related fields confess to a pronounced feeling of disappointment in what has been achieved in the past 25 years. Workers entered the field around 1950, and even around 1960, with high hopes that are very far from having been realized in 1972. In no part of the field have the discoveries made so far produced the major impact that was then promised...

One rather general cause for the disappointments that have been experienced: failure to recognize the implications of the 'combinatorial explo-

reserved for that pattern, but is distributed over the connections among a large number of processing units." (p. 33)

This led directly to Rosenblatt's idea that such machines should be able to acquire their ability through learning rather than by being programmed with features and rules:

"If the knowledge is in the strengths of the connections, learning must be a matter of finding the right connection strengths so that the right patterns of activation will be produced under the right circumstances. This is an extremely important property of this class of models, for it opens up the possibility that an information processing mechanism could learn, as a result of tuning its connections, to capture the interdependencies between activations that it is exposed to in the course of processing." (p. 32)

David E. Rumelhart, James L. McClelland, and the PDP Research Group, *Parallel Distributed Processing. Vol 1.* (Cambridge: Bradford/MIT Press, 1986), p. 158.

[17]F. Rosenblatt, *Mechanisation of Thought Processes: Proceedings of a Symposium held at the National Physical Laboratory, November 1958. Vol. 1.*, p. 449., (London: HM Stationery Office).

[18]Marvin Minsky and Seymour Papert, *Perceptrons: An Introduction to Computational Geometry*, (Cambridge: The MIT Press, 1969), p. 19.

sion'. This is a general obstacle to the construction of a... system on a large knowledge base which results from the explosive growth of any combinatorial expression, representing numbers of possible ways of grouping elements of the knowledge base according to particular rules, as the base's size increases.[19]

As David Rumelhart succinctly sums it up: "Combinatorial explosion catches you sooner or later, although sometimes in different ways in parallel than in serial."[20] Both sides had, as Jerry Fodor once put it, walked into a game of three-dimensional chess thinking it was tic-tac-toe. Why then, so early in the game, with so little known and so much to learn, did one team of researchers triumph at the total expense of the other? Why, at this crucial branchpoint, did the symbolic representation project become the only game in town?

Everyone who knows the history of the field will be able to point to the proximal cause. About 1965 Minsky and Papert, who were running a laboratory at MIT dedicated to the symbol manipulation approach and therefore competing for support with the perceptron projects, began circulating drafts of a book directly attacking perceptrons. In the book they made clear their scientific position:

Perceptrons have been widely publicized as "pattern recognition" or "learning" machines and as such have been discussed in a large number of books, journal articles, and voluminous "reports". Most of this writing... is without scientific value.[21]

But their attack was also a philosophical crusade. They rightly saw that traditional reliance on reduction to logical primitives was being challenged by a new holism.

Both of the present authors (first independently and later together) became involved with a somewhat therapeutic compulsion: to dispel what we feared to be the first shadows of a "holistic" or "Gestalt" misconception that would threaten to haunt the fields of engineering and artificial intelligence as it had earlier haunted biology and psychology.[22]

They were quite right. Artificial neural nets may, but need not, allow an interpretation of their hidden nodes in terms of features a human being could recognize and use to solve the problem. While neural network modeling itself is committed to neither view, it can be demonstrated that association does not *require* that the hidden nodes be interpretable. Holists like Rosenblatt happily assumed that individual nodes or patterns of nodes were not picking out fixed features of the domain.

Minsky and Papert were so intent on eliminating all competition and so secure in the atomistic tradition that runs from Descartes to early Wittgenstein, that the book suggests much more than it actually demonstrates. They set out to analyze the capacity of a one-layer perceptron while completely ignoring in the mathematical portion of their book Rosenblatt's chapters on multilayer machines and his proof of the convergence of an (inefficient) probabilistic learning algorithm based on back propagation of errors.[23] According to Rumelhart and McClelland:

Minsky and Papert set out to show which functions can and cannot be computed by one-layer machines. They demonstrated, in particular, that such perceptrons are unable to calculate such mathematical functions as parity (whether an odd or even number of points are on in the retina) or the topological function of connectedness (whether all points that are on are connected to all other points that are on either directly or via other points that are also on) without making use of absurdly large numbers of predicates. The analysis is extremely elegant and demonstrates the importance of a mathematical approach to analyzing computational systems.[24]

[19]Sir James Lighthill, "Artificial Intelligence: A General Survey" in *Artificial Intelligence: a paper symposium*, (London: Science Research Council, 1973).

[20]David E. Rumelhart, James L. McClelland, op. cit., p. 158.

[21]Minsky and Papert, *Perceptrons*, p. 4.

[22]Ibid., p. 19.

[23]F. Rosenblatt, *Principles of Neurodynamics, Perceptrons and the Theory of Brain Mechanisms*, (Washington, D.C.: Spartan Book, 1962), p. 292. See also:

"The addition of a fourth layer of signal transmission units, or cross-coupling the A-units of a three-layer perceptron, permits the solution of generalization problems, over arbitrary transformation groups." (p.576)

"In back-coupled perceptrons, selective attention to familiar objects in a complex field can occur. It is also possible for such a perceptron to attend selectively to objects which move differentially relative to their background." (p. 576)

[24]Rumelhart and McClelland, op. cit., p. 111.

But the implications of the analysis are quite limited. Rumelhart and McClelland continue:

Essentially... although Minsky and Papert were exactly correct in their analysis of the *one-layer perceptron*, the theorems don't apply to systems which are even a little more complex. In particular, it doesn't apply to multilayer systems nor to systems that allow feedback loops.[25]

Yet, in the conclusion to *Perceptrons*, when Minsky and Papert ask themselves the question: "Have you considered perceptrons with many layers?", they give the impression, while rhetorically leaving the question open, of having settled it.

Well, we have considered Gamba machines, which could be described as "two layers of perceptron." We have not found (by thinking or by studying the literature) any other really interesting class of multilayered machine, at least none whose principles seem to have a significant relation to those of the perceptron... We consider it to be an important research problem to elucidate (or reject) our intuitive judgment that the extension is sterile.[26]

Their attack of gestalt thinking in A.I. succeeded beyond their wildest dreams. Only an unappreciated few, among them S. Grossberg, J.A. Anderson and T. Kohonen, took up the "important research problem". Indeed, almost everyone in AI assumed that neural nets had been laid to rest forever. Rumelhart and McClelland note:

Minsky and Papert's analysis of the limitations of the one-layer perceptron, coupled with some of the early successes of the symbolic processing approach in artificial intelligence, was enough to suggest to a large number of workers in the field that there was no future in perceptron-like computational devices for artificial intelligence and cognitive psychology.[27]

But why was it enough? Both approaches had produced some promising work and some unfounded promises.[28] It was too early to close accounts on either approach. Yet something in Minsky and Papert's book struck a responsive chord. It seemed AI workers shared the quasi-religious philosophical prejudice against holism which motivated the attack. One can see the power of the tradition, for example, in Newell and Simon's article on physical symbol systems. The article begins with the scientific hypothesis that the mind and the computer are intelligent by virtue of manipulating discrete symbols, but it ends with a revelation. "The study of logic and computers has revealed to us that intelligence resides in physical-symbol systems."[29]

Holism could not compete with such intense philosophical convictions. Rosenblatt was discredited along with the hundreds of less responsible network research groups that his work had encouraged. His research money dried up, he had troubled getting his work published, he became depressed, and one day his boat was found empty at sea. Rumor had it that he had committed suicide. Whatever the truth of that rumor, one thing is certain: by 1970, as far as AI was concerned, neural nets were dead. Newell, in his history of AI, says the issue of symbols versus numbers "is certainly not alive now and has not been for a long time."[30] Rosenblatt is not even mentioned in John Haugeland's or in Margaret Boden's histories of the AI field.[31]

[25]Ibid., p. 112.

[26]Minsky and Papert, op. cit., pp. 231-232.

[27]Rumelhart and McClelland, op. cit., p. 112.

[28]For an evaluation of the symbolic representation approach's actual successes up to 1970, see H. Dreyfus, *What Computers Can't Do*, (New York: Harper and Row, 2nd edition, 1979).

[29]Newell and Simon, "Computer Science and Empirical Inquiry", op. cit., p. 64.

[30]Op. cit., p. 10.

[31]J. Haugeland, *Artificial Intelligence: The Very Idea*, (Cambridge: Bradford/MIT Press, 1985). M. Boden, *Artificial Intelligence and Natural Man*, (New York: Basic Books, 1977). Work on neural nets was continued in a marginal way in psychology and neuro-science. James A. Anderson at Brown University continued to defend a net model in psychology, although he had to live off other researchers' grants, and Stephen Grossberg worked out an elegant mathematical implementation of elementary cognitive capacities. For Anderson's position see, "Neural Models with Cognitive Implications" in *Basic Processing in Reading*, D. LaBerse and S.J. Samuels edts., (New Jersey: Erlbaum, 1978). For examples of Grossberg's work during the dark ages, see his book *Studies of Mind and Brain: Neural Principles of learning, perception, development, cognition and motor control*, (Boston: Reidel Press, 1982). Kohonen's early work is reported in *Associative Memory - A System-Theoretical approach*, (Berlin: Springer Verlag, 1977).

At M.I.T. Minsky continued to lecture on neural nets and assign theses investigating their logical properties. But, according to Papert, this was only because nets *had* interesting mathematical properties whereas nothing interesting could be proved concerning the properties of symbol systems. Moreover, many A.I. researchers assumed

But blaming the rout of the connectionists on an anti-holistic prejudice is too simple. There was a deeper way philosophical assumptions influenced intuition and led to an overestimation of the importance of the early symbol processing results. The way it looked at the time was that the perceptron people had to do an immense amount of mathematical analysis and calculating to solve even the most simple problems of pattern recognition such as discriminating horizontal from vertical lines in various parts of the receptive field, while the symbol manipulating approach had relatively effortlessly solved hard problems in cognition such as proving theorems in logic and solving puzzles such as the cannibal-missionary problem. Even more importantly, it seemed that given the computing power available at the time, the neural net researchers could only do speculative neuro-science and psychology, while the simple programs of symbolic representationists were on their way to being useful. Behind this way of sizing up the situation was the assumption that thinking and pattern recognition are two distinct domains and that thinking is the more important of the two. As we shall see later in our discussion of the common sense knowledge problem, this way of looking at things ignores both the preeminent role of pattern discrimination in human expertise and also the background of common sense understanding which is presupposed in real world, everyday thinking. Taking account of this background may well require pattern recognition.

This gets us back to the philosophical tradition. It was not just Descartes and his descendants which stood behind symbolic information processing, but all of Western philosophy. According to Heidegger, traditional philosophy is defined from the start by its focusing on facts in the world while "passing over" the world as such.[32] This means that philosophy has from the start systematically ignored or distorted the everyday context of human activity.[33] That branch of the philosophical tradition that descends from Socrates, to Plato, to Descartes, to Leibniz, to Kant, to conventional AI takes it for granted, in addition, that understanding a domain consists in having a *theory* of that domain. A theory formulates the relationships between objective, *context-free* elements (simples, primitives, features, attributes, factors, data points, cues, etc.) in terms of abstract principles (covering laws, rules, programs, etc.).

Plato held that in theoretical domains such as mathematics and perhaps ethics, thinkers apply explicit, context-free rules or theories they learned in another life, outside the everyday world. Once learned, such theories function in this world by controlling the thinker's mind whether he is conscious of them or not. Plato's account did not apply to everyday skills but only to domains in which there is a *priori* knowledge. The success of theory in the natural sciences, however, reinforced the idea that in any orderly domain there must be some set of context-free elements and some abstract relations between those elements which accounts for the order of that domain and for man's ability to act intelligently in it. Thus Leibniz boldly generalized the rationalist account to all forms of intelligent activity, even everyday practice.

The most important observations and turns of skill in all sorts of trades and professions are as yet unwritten. This fact is proved by experience when passing from theory to practice we desire to accomplish something. *Of course, we can also write up this practice, since it is at bottom just another theory more complex and particular...*[34]

The symbolic information processing approach gains its assurance from this transfer to all domains of methods that were developed by philosophers and which have succeeded in the natural sciences. Since, on this view, any domain must be formalizable, the way to do AI in any area

that since Turing Machines were symbol manipulators and Turing had proved that Turing Machines could compute anything, he had proved that all intelligibility could be captured by logic. On this view a holistic (and in those days statistical) approach needed justification while the symbolic A.I. approach did not. This confidence, however, was based on confusing the uninterpreted symbols (zeroes and ones) of a Turing Machine with the semantically interpreted symbols of A.I.

[32]Martin Heidegger, *Being and Time*, (New York: Harper and Row), 1962, Sections 14-21, See H. Dreyfus, *Being-in-the-world: A Commentary on Division I of Being and Time*, (Cambridge: MIT Press/Bradford Books, 1988).

[33]According to Heidegger, Aristotle came closer than any other philosopher to understanding the importance of everyday activity, but even he succombed to the philosophical distortions of the phenomenon of the everyday world implicit in common sense.

[34]Leibniz, *Selections*, op. cit., p. 48 (Our italics.)

is obviously to find the context-free elements and principles and base a formal, symbolic representation on this theoretical analysis. Terry Winograd characteristically describes his AI work in terms borrowed from physical science:

We are concerned with developing a formalism, or "representation," with which to describe... knowledge. We seek the "atoms" and "particles" of which it is built, and the "forces" that act on it.[35]

No doubt theories about the universe are often built up gradually by modeling relatively simple and isolated systems and then making the model gradually more complex and integrating it with models of other domains. This is possible because all the phenomena are presumably the result of the law-like relations between what Papert and Minsky call "structural primitives." Since no one argues for atomistic reductionism in A.I. it seems that A.I. workers must implicitly assume that the abstraction of elements from their everyday context, which defines philosophy and works in natural science, must also work in AI. This would account for the way the physical symbol system hypothesis so quickly turned into a revelation and for the ease with which Papert's and Minsky's book triumphed over the holism of the perceptron.

Teaching philosophy at M.I.T. in the mid-sixties, Hubert was soon drawn into the debate over the possibility of AI. It was obvious to him that researchers such as Newell, Simon, and Minsky were the heirs to the philosophical tradition. But given his understanding of later Wittgenstein and early Heidegger, that did not seem to be a good omen for the reductionist research program. Both these thinkers had called into question the very tradition on which symbolic information processing was based. Both were holists, both were struck by the importance of everyday practices, and both held that one could not have a theory of the everyday world.

It is one of the ironies of intellectual history that Wittgenstein's devastating attack on his own *Tractatus*, his *Philosophical Investigations*,[36] was published in 1953 just as AI took over the abstract, atomistic tradition he was attacking. After writing the *Tractatus* Wittgenstein spent years doing what he called "phenomenology"[37] – looking in vain for the atomic facts and basic objects his theory required. He ended by abandoning his *Tractatus* and all rationalistic philosophy. He argued that the analysis of everyday situations into facts and rules (which is where most traditional philosophers and AI researchers think theory must begin) is itself only meaningful in some context and for some purpose. Thus the elements chosen already reflect the goals and purposes for which they are carved out. When we try to find the ultimate context-free, purpose-free elements, as we must if we are going to find the primitive symbols to feed a computer, we are in effect trying to free aspects of our experience of just that pragmatic organization which makes it possible to use them intelligibly in coping with everyday problems.

In the *Philosophical Investigations* Wittgenstein directly criticizes the logical atomism of the *Tractatus*.

"What lies behind the idea that names really signify simples"? – Socrates says in the *Theaetetus*: "If I make no mistake, I have heard some people say this: there is no definition of the primary elements – so to speak – out of which we and everything else are composed... But just as what consists of these primary elements is itself complex, so the names of the elements become descriptive language by being compounded together." Both Russell's 'individuals' and my 'objects' (*Tractatus Logico-Philosophicus*) were such primary elements. But what are the simple constituent parts of which reality is composed?... It makes no sense at all to speak absolutely of the 'simple parts of a chair.'[38]

Already in the 1920s Martin Heidegger had reacted in a similar way against his mentor, Edmund Husserl, who regarded himself as the culmination of the Cartesian tradition and was, therefore, the grandfather of AI.[39] Husserl argued that an act of consciousness or *noesis* does not,

[35]T. Winograd, "Artificial Intelligence and Language Comprehension," in *Artificial Intelligence and Language Comprehension*, National Institute of Education, 1976, p. 9.

[36]Wittgenstein, *Philosophical Investigations*, (Oxford: Basil Blackwell, 1953).

[37]Ludwig Wittgenstein, *Philosophical Remarks*, University of Chicago Press, 1975.

[38]Wittgenstein, *Philosophical Investigations*, (Oxford: Basil Blackwell, 1953), p. 21.

[39]See H. Dreyfus ed., *Husserl, Intentionality and Cognitive Science*, (Cambridge: MIT Press/Bradford Books, 1982).

on its own, grasp an object; rather, the act has intentionality (directedness) only by virtue of an "abstract form" or meaning in the *noema* correlated with the act.[40]

This meaning or symbolic representation, as conceived by Husserl, was a complex entity that had a difficult job to perform. In *Ideas*[41] Husserl bravely tries to explain how the *noema* gets the job done. Reference is provided by predicate-*senses* which, like Fregean *Sinne*, just have the remarkable property of picking out objects' atomic properties. These predicates are combined into complex "descriptions" of complex objects, as in Russell's theory of descriptions. For Husserl, who is close to Kant on this point, the *noema* contains a hierarchy of strict rules. Since Husserl thought of intelligence as a context-determined, goal-directed activity, the mental representation of any type of object had to provide a context or "horizon" of expectations or "predelineations" for structuring the incoming data: "a rule governing *possible* other consciousness of the object as identical – possible, as exemplifying essentially predelineated types."[42] The *noema* must contain a rule describing all the features which can be expected with certainty in exploring a certain *type* of object–features which remain "inviolably the same: as long as the objectivity remains intended as *this* one and of this kind."[43] The rule must also prescribe "predelineations" of properties that are possible but not necessary features of this type of object: "Instead of a completely determined sense, there is always, therefore, a *frame of empty sense...*"[44]

In 1973 Marvin Minsky proposed a new data structure, remarkably similar to Husserl's, for representing everyday knowledge:

> A *frame* is a data-structure for representing a stereotyped situation, like being in a certain kind of living room, or going to a child's birthday party...
>
> We can think of a frame as a network of nodes and relations. The top levels of a frame are fixed, and represent things that are always true about the supposed situation. The lower levels have many *terminals* – slots that must be filled by specific instances or data. Each terminal can specify conditions its assignments must meet...
>
> Much of the phenomenological power of the theory hinges on the inclusion of expectations and other kinds of presumptions. *A frame's terminals are normally already filled with "default" assignments.*[45]

In Minsky's model of a frame, the "top level" is a developed version of what in Husserl's terminology remains "inviolably the same" in the representation, and Husserl's predelineations have become "default assignments" – additional features that can normally be expected. The result is a step forward in AI techniques from a passive model of information-processing to one which tries to take account of the interactions between a knower and the world. The task of AI thus converges with the task of transcendental phenomenology. Both must try in everyday domains to find frames constructed from a set of primitive predicates and their formal relations.

Heidegger, before Wittgenstein, carried out, in response to Husserl, a phenomenological description of the everyday world and everyday objects like chairs and hammers, and like Wittgenstein he found that the everyday world could not be represented by a set of context-free elements. It was Heidegger who forced Husserl to face precisely this problem. He pointed out that there are other ways of "encountering" things than relating to them as objects defined by a set of predicates. When we use a piece of equipment like a hammer, Heidegger pointed out, we actualize a skill (which need not be represented in the mind) in the context of a socially organized nexus of equipment, purposes, and human roles (which need not be represented as a set of facts). This context or world, and our everyday ways of skillful coping in it which Heidegger called *circumspection*, are not

[40] "Der Sinn... so wie wir ihn bestimmt haben, ist nicht ein konkretes Wesen im Gesamtbestande des Noema, sondern eine Art ihm einwohnender abstrackter Form." Edmund Husserl, *Ideen Zu Einer Reinen Phänomenologie und Phänomenologischen Philosophie*, Nijhoff, 1950. For textual evidence that Husserl held that the noema accounts for the intentionality of mental activity, see H. Dreyfus, "Husserl's Perceptual Noema" in *Husserl, Intentionality and Cognitive Science*, M.I.T./Bradford Books, 1982.

[41] E. Husserl, *Ideas Pertaining to a Pure Phenomenology and to a Phenomenological Philosophy*, trans. F. Kersten, (The Hague: Nijhoff, 1982).

[42] Edmund Husserl, *Cartesians Meditations*, trans. D. Cairns, (The Hague: Nijhoff, 1960) p. 45

[43] Ibid., p. 53.

[44] Ibid., p. 51.

[45] Marvin Minsky, "A Framework for Representing Knowledge," in *Mind Design*, J. Haugeland, ed., p. 96.

something we *think* but, as part of our socialization, forms the way we *are*. Heidegger concluded:

The context... can be taken formally in the sense of a system of relations. But... the phenomenal content of these 'relations' and 'relata'... is such that they resist any sort of mathematical functionalization; nor are they merely something thought, first posited in an 'act of thinking.' They are rather relationships in which concernful circumspection as such already dwells.[46]

This defines the splitting of the ways between Husserl and AI on the one hand, and Heidegger and later Wittgenstein on the other. The crucial question becomes: Can there be a theory of the everyday world as rationalist philosophers have always held? Or is the common sense background rather a combination of skills, practices, discriminations, etc., which are not intentional states, and so, *a fortiori*, do not have any representational content to be explicated in terms of elements and rules?

Husserl tried to avoid the problem posed by Heidegger by making a move soon to become familiar in AI circles. He claimed that the world, the background of significance, the everyday context, was merely a very complex system of facts correlated with a complex system of beliefs, which, since they have truth conditions, he called "validities". Thus one could, in principle, suspend one's dwelling in the world and achieve a detached, description of the human belief system. One could thus complete the task that had been implicit in philosophy since Socrates. One could make explicit the beliefs and principles underlying all intelligent behavior. As Husserl put it:

Even the background... of which we are always concurrently conscious but which is momentarily irrelevant and remains completely unnoticed, still functions according to its implicit validities.[47]

Since he firmly believed that the shared background could be made explicit as a belief system Husserl was ahead of his time in raising the question of the possibility of AI. After discussing the possibility of a formal axiomatic system describing experience, and pointing out that such a system of axioms and primitives – at least as we know it in geometry – could not describe everyday shapes such as "scalloped" and "lens-shaped," Husserl leaves open the question whether these everyday concepts could nonetheless be formalized. (This is like raising and leaving open the A.I. question whether one can axiomatize common sense physics.) Picking up Leibniz's dream of a mathesis of all experience, Husserl remarks:

The pressing question is ... whether there could not be... an idealizing procedure that substitutes pure and strict ideals for intuited data and that would... serve... as the basic medium for a mathesis of experience.[48]

But, as Heidegger predicted, the task of writing out a complete theoretical account of everyday life turned out to be much harder than initially expected. Husserl's project ran into serious trouble, and there are signs that Minsky's has too. During twenty-five years of trying to spell out the components of the subject's representation of everyday objects, Husserl found that he had to include more and more of a subject's common-sense understanding of the everyday world:

To be sure, even the tasks that present themselves when we take single types of objects as restricted clues prove to be extremely complicated and always lead to extensive disciplines when we penetrate more deeply. That is the case, for example, with... spatial objects (to say nothing of a Nature) as such, of psycho-physical being and humanity as such, culture as such.[49]

He spoke of the noema's "huge concreteness"[50] and of its "tremendous complication,"[51] and he sadly concluded at the age of seventy-five that he was a perpetual beginner and that phenomenology was an "infinite task."[52]

There are hints in his frame paper that Minsky has embarked on the same "infinite task" that eventually overwhelmed Husserl:

Just constructing a knowledge base is a major intellectual research problem... We still know far too little about the contents and struc-

[46]Heidegger, op. cit., pp. 121-122.

[47]Edmund Husserl, *Crisis of European Sciences and Transcendental Phenomenology*, trans. D. Carr, (Evanston: Northwestern University Press, 1970), p. 149.

[48]Husserl, *Ideen zu einer reinen Phanomenologie und phenomenologischen Philosophie*, Drittes Buch, 1913, #75, p. 134.

[49]Husserl, *Cartesians Meditations*, pp. 54-55.

[50]Husserl, *Formal and Transcendental Logic*, trans. D. Cairns (The Hague: Nijhoff, 1969), p. 244.

[51]Ibid., p. 246.

[52]Husserl, *Crisis*, p. 291.

ture of common-sense knowledge. A "minimal" common-sense system must "know" something about cause-effect, time, purpose, locality, process, and types of knowledge... We need a serious epistemological research effort in this area.[53]

To a student of contemporary philosophy Minsky's naivete and faith were astonishing. Husserl's phenomenology *was* just such a research effort. Indeed, philosophers, from Socrates to Leibniz, to early Wittgenstein, had carried on serious epistemological research in this area for two thousand years without notable success.

In the light of Wittgenstein's reversal and Heidegger's devastating critique of Husserl, Hubert predicted trouble for symbolic information processing. As Newell notes in his history of AI, Hubert's warning was ignored:

Dreyfus's central intellectual objection... is that the analysis of the context of human action into discrete elements is doomed to failure. This objection is grounded in phenomenological philosophy. Unfortunately, this appears to be a nonissue as far as AI is concerned. The answers, refutations, and analyses that have been forthcoming to Dreyfus's writings have simply not engaged this issue – which indeed would be a novel issue if it were to come to the fore.[54]

The trouble was not long in coming to the fore, however, as the everyday world took its revenge on AI as it had on traditional philosophy. As we see it, the research program launched by Newell and Simon has gone through three ten-year stages. From 1955-1965 two research themes, representation and search, dominated the field then called Cognitive Simulation. Newell and Simon showed, for example, how a computer could solve the cannibal and missionary problem, using the general heuristic search principle known as means-end analysis, viz. use any available operation that reduces the distance between the description of the current situation and the description of the goal. They then abstracted this heuristic technique and incorporated it into their General Problem Solver (GPS).

The second stage (1965-1975), led by Marvin Minsky and Seymour Papert at M.I.T., was concerned with what facts and rules to represent. The idea was to develop methods for dealing systematically with knowledge in isolated domains called micro-worlds. Famous programs written around 1970 at M.I.T. include Terry Winograd's SHRDLU which could obey commands given in a subset of natural language about a simplified blocks-world, Thomas Evan's Analogy Problem Program, David Waltz's Scene Analysis Program and Patrick Winston's program which learned concepts from examples.

The hope was that the restricted and isolated "micro-worlds" could be gradually made more realistic and combined so as to approach real world understanding. But researchers confused two domains which, following Heidegger, we shall distinguish as universe and world. A set of interrelated facts may constitute a *universe*, like the physical universe, but it does not constitute a *world*. The latter, like the world of business, the world of theater, or the world of the physicist, is an organized body of objects, purposes, skills, and practices on the basis of which human activities have meaning or make sense. To see the difference one can contrast the *meaningless* physical *universe* with the *meaningful world* of the discipline of physics. The world of physics, the business world, and the theater world, make sense only against a background of common human concerns. They are local elaborations of the one common-sense world we all share. That is, sub-worlds are not related like isolable physical systems to larger systems they *compose*, but are rather, local elaborations of a whole, which they *presuppose*. Micro-worlds were *not* worlds but isolated meaningless domains, and it has gradually become clear that there was no way they could be combined and extended to arrive at the world of everyday life.

In its third and so far final stage, roughly from 1975 to the present, AI has been wrestling with what has come to be called the common-sense knowledge problem. The representation of knowledge was always a central problem for work in AI, but the two earlier periods – cognitive simulation and micro-worlds – were characterized by an attempt to avoid the problem of common-sense knowledge by seeing how much could be done with as little knowledge as possible. By the middle 1970s, however, the issue had to be faced. Various data structure such as Minsky's frames and Roger Schank's scripts have been tried without

[53]Minsky, op. cit., p. 124.

[54]Newell, "Intellectual Issues in the history of Artificial Intelligence," p. 222-223.

success. The common-sense knowledge problem has kept AI from even beginning to fulfill Simon's prediction made twenty years ago, that "within twenty years machines will be capable of doing any work a man can do."[55]

Indeed, the common-sense knowledge problem has blocked all progress in theoretical AI for the past decade. Winograd was one of the first to see the limitations of SHRDLU and all script and frame attempts to extend the micro-worlds approach. Having "lost faith" in AI, he now teaches Heidegger in his computer science courses at Stanford, and points out "the difficulty of formalizing the common-sense background that determines which scripts, goals and strategies are relevant and how they interact."[56]

What sustains AI in this impasse is the conviction that the common sense knowledge problem must be solvable since human beings have obviously solved it. But human beings may not normally use common sense *knowledge* at all. As Heidegger and Wittgenstein point out, what common sense *understanding* amounts to might well be *everyday know-how*. By know-how we do not mean procedural rules, but knowing what to do in a vast number of special cases.[57] For example, common sense physics has turned out to be extremely hard to spell out in a set of facts and rules. When one tries, one either requires more common sense to understand the facts and rules one finds or else one produces formulas of such complexity that it seems highly unlikely they are in a child's mind.

Doing theoretical physics also requires background skills which may not be formalizable, but the domain itself can be described by abstract laws that make no reference to these background skills. AI researchers conclude that common sense physics too must be expressible as a set of abstract principles. But it just may be that the problem of finding a *theory* of common sense physics is insoluble because the domain has no theoretical structure. By playing all day with all sorts of liquids and solids for several years the child may simply have learned to discriminate prototypical cases of solids, liquids, etc. and learned typical skilled responses to their typical behavior in typical circumstances. The same might well be the case for the social world. If background understanding is indeed a skill, and skills are based on whole patterns and not on rules, we would expect symbolic representations to fail to capture our common-sense understanding.

In the light of this impasse, classical, symbol-based AI appears more and more to be a perfect example of what Imre Lakatos has called a degenerating research program.[58] As we have seen, AI began auspiciously with Newell and Simon's work at RAND, and by the late 1960s had turned into a flourishing research program. Minsky predicted that "within a generation the problem of creating 'artificial intelligence' will be substantially solved."[59] Then, rather suddenly, the field ran into unexpected difficulties. It turned out to be much harder than one expected to formulate a theory of common-sense. It was not, as Minsky had hoped, just a question of cataloguing a few hundred thousand facts. The common-sense knowledge problem became the center of concern. Minsky's mood changed completely in five years. He told a reporter: "the AI problem is one of the hardest science has ever undertaken."[60]

The Rationalist tradition had finally been put to an empirical test and it had failed. The idea of producing a formal, atomistic theory of the everyday common-sense world and representing that theory in a symbol manipulator had run into just the difficulties Heidegger and Wittgenstein discovered. Frank Rosenblatt's intuition that it would be hopelessly hard to formalize the world and thus give a formal specification of intelligent behavior had been vindicated. His repressed research program – using the computer to instantiate a holistic model of an idealized brain – which had never really been refuted, became again a live option.

In journalistic accounts of the history of AI Rosenblatt is vilified by anonymous detractors as a snake-oil salesman:

[55]H. Simon, *The Shape of Automation for Men and Management*, Harper and Row, 1965, p. 96.

[56]T. Winograd, "Computer Software for Working with Language," *Scientific American*, September 1984, p. 142.

[57]This account of skill is spelled out and defended in Hubert and Stuart Dreyfus, *Mind Over Machine*, (New York: Free Press/Macmillan, 1986).

[58]Imre Lakatos, *Philosophical Papers*, ed. J. Worrall, (Cambridge: Cambridge University Press, 1978).

[59]Minsky, *Computation: Finite and Infinite Machines*, (New York: Prentice Hall, 1977), p. 2.

[60]Gina Kolata, "How Can Computers Get Common Sense?", *Science*, Vol. 217, 24 September 1982. p. 1237.

Present-day researchers remember that Rosenblatt was given to steady and extravagant statements about the performance of his machine. "He was a press agent's dream," one scientist says, "a real medicine man. To hear him tell it, the Perceptron was capable of fantastic things. And maybe it was. But you couldn't prove it by the work Frank did."[61]

In fact he was much clearer about the capacities and limitations of the various types of perceptrons than Simon and Minsky were about their symbolic programs.[62] Now he is being rehabilitated. Rumelhart, Hinton and McClelland reflect this new appreciation of his pioneering work:

Rosenblatt's work was very controversial at the time, and the specific models he proposed were not up to all the hopes he had for them. But his vision of the human information processing system as a dynamic, interactive, self-organizing system lies at the core of the PDP approach.[63]

The studies of perceptrons... clearly anticipated many of the results in use today. The critique of perceptrons by Minsky and Papert was widely misinterpreted as destroying their credibility, whereas the work simply showed limitations on the power of the most limited class of perceptron-like mechanisms, and said nothing about more powerful, multiple layer models.[64]

Frustrated AI researchers, tired of clinging to a research program which Jerry Lettvin characterized in the early 1980s as "the only straw afloat", flocked to the new paradigm. Rumelhart and McClelland's book, *Parallel Distributed Processing*, sold 6000 copies the day it went on the market. 30,000 are now in print. As Paul Smolensky put it:

In the past half-decade the connectionist approach to cognitive modeling has grown from an obscure cult claiming a few true believers to a movement so vigorous that recent meetings of the Cognitive Science Society have begun to look like connectionist pep rallies.[65]

If multilayered networks succeed in fulfilling

[61]Pamela McCorduck, *Machines Who Think*, (San Francisco: W.H. Freeman and Company, 1979), p. 87.

[62]Some typical quotations from Rosenblatt's *Principles of Neurodynamics*:

"In a learning experiment, a perceptron is typically exposed to a sequence of patterns containing representatives of each type or class which is to be distinguished, and the appropriate choice of a response is "reinforced" according to some rule for memory modification. The perceptron is then presented with a test stimulus, and the probability of giving the appropriate response for the class of the stimulus is ascertained... If the test stimulus activates a set of sensory elements which are entirely distinct from those which were activated in previous exposures to stimuli of the same class, the experiment is a test of "pure generalization". The simplest of perceptrons... have no capability for pure generalization, but can be shown to perform quite respectably in discrimination experiments particularly if the test stimulus is nearly identical to one of the patterns previously experienced." (p. 68)

"Perceptrons considered to date show little resemblance to human subjects in their figure-detection capabilities, and gestalt-organizing tendencies." (p. 71)

"The recognition of sequences in rudimentary form is well within the capability of suitably organized perceptrons, but the problem of figural organization and segmentation presents problems which are just as serious here as in the case of static pattern perception." (p. 72)

"In a simple perceptron, patterns are recognized before "relations"; indeed, abstract relations, such as "A is above B" or "the triangle is inside the circle" are never abstracted as such, but can only be acquired by means of a sort of exhaustive rote-learning procedure, in which every case in which the relation holds is taught to the perceptron individually." (p. 73)

"A network consisting of less than three layers of signal transmission units, or a network consisting exclusively of linear elements connected in series, is incapable of learning to discriminate classes of patterns in an isotropic environment (where any pattern can occur in all possible retinal locations, without boundaries effects)." (p. 575)

"A number of speculative models which are likely to be capable of learning sequential programs, analysis of speech into phonemes, and learning substantive "meanings" for nouns and verbs with simple sensory referents have been presented in the preceding chapters. Such systems represent the upper limits of abstract behavior in perceptrons considered to date. They are handicapped by a lack of a satisfactory "temporary memory", by an inability to perceive abstract topological relations in a simple fashion, and by an inability to isolate meaningful figural entities, or objects, except under special conditions" (p. 577).

"The applications most likely to be realizable with the kinds of perceptrons described in this volume include character recognition and "reading machines", speech recognition (for distinct, clearly separated words), and extremely limited capabilities for pictorial recognition, or the recognition of objects against simple backgrounds. "Perception" in a broader sense may be potentially within the grasp of the descendants of our present models, but a great deal of fundamental knowledge must be obtained before a sufficiently sophisticated design can be prescribed to permit a perceptron to compete with a man under normal environmental conditions." (p. 583)

[63]D. Rumelhart and J. McClelland, op. cit., Vol 1., p. 45.

[64]Ibid., Vol. 2., p. 535.

[65]Paul Smolensky, "On the proper treatment of connectionism", *Behavioral and Brain Sciences*, final draft, p. 1, summer 1987.

their promise researchers will have to give up Descartes', Husserl's and early Wittgenstein's conviction that the only way to produce intelligent behavior is to mirror the world with a formal theory in the mind. Worse, one may have to give up the more basic intuition at the source of philosophy that there must be a theory of every aspect of reality, i.e., there must be elements and principles in terms of which one can account for the intelligibility of any domain. Neural networks may show that Heidegger, later Wittgenstein and Rosenblatt were right in thinking that we behave intelligently in the world without having a theory of that world. If a theory is not *necessary* to explain intelligent behavior we have to be prepared to raise the question whether, in everyday domains, such a theoretical explanation is even *possible*.

Neural net modelers, influenced by symbol manipulating AI, are expending considerable effort, once their nets have been trained to perform a task, trying to find the features represented by individual nodes and sets of nodes. Results thus far are equivocal. Consider Geoffrey Hinton's network for learning concepts by means of distributed representations.[66] Hinton's network can be trained to encode relationships in a domain which human beings conceptualize in terms of features, without the network being given the features that human beings use. Hinton produces examples of cases in which in the trained network some nodes can be interpreted as corresponding to the features that human beings pick out, although they only roughly correspond to these features. Most nodes, however, cannot be interpreted semantically at all. A feature used in a symbolic representation is either present or not. In the net, however, although certain nodes are more active when a certain feature is present in the domain, the amount of activity varies not just with the presence or absence of this feature, but is affected by the presence or absence of other features as well.

Hinton has picked a domain, family relationships, which is constructed by human beings precisely in terms of the features, such as generation and nationality, which human beings normally notice. Hinton then analyzes those cases in which, starting with certain random initial connection strengths, some nodes after learning can be interpreted as representing these features. Calculations using Hinton's model show, however, that even his net seems, for some random initial connection strengths, to learn its associations without any obvious use of these everyday features.

In one very limited sense, any successfully trained multilayer net has an interpretation in terms of features – not everyday features but what we shall call highly abstract features. Consider the particularly simple case of layers of binary units activated by feedforward, but not lateral or feedback, connections. To construct an account from a network that has learned certain associations, each node one level above the input nodes could, on the basis of connections to it, be interpreted as detecting when one of a certain set of input patterns is present. (Some of the patterns will be the ones used in training and some will never have been used.) If the set of input patterns which a particular node detects is given an invented name (it almost certainly won't have a name in our vocabulary), the node could be interpreted as detecting the highly abstract feature so named. Hence, every node one level above the input level can be characterized as a feature detector. Similarly, every node a level above these nodes can be interpreted as detecting a higher-order feature which is defined as the presence of one of a specified set of patterns among the first level features detectors. And so on up the hierarchy.

The fact that intelligence, defined as the knowledge of a certain set of associations appropriate to a domain, can always be accounted for in terms of relations among a number of highly abstract features of a skill domain does not, however, preserve the rationalist intuition that these explanatory features must capture the essential structure of the domain, i.e., that one could base a theory on them. If the net is taught one more association of an input/output pair (where the input prior to training produces an output different from the one to be learned), the interpretation of at least some of the nodes will have to be changed. So the features which some of the nodes picked out before the last instance of training would turn out not to have been invariant structural features of

[66] Geoffrey Hinton, "Learning Distributed Representations of Concepts," *Proceedings of the 8th Annual Conference Cognitive Science Society*, Amherst, Mass., Aug. 1986

the domain.

Once one has abandoned the philosophical approach of classical AI and accepted the atheoretical claim of neural netmodeling, one question remains: How much of everyday intelligence can such a network be expected to capture? Classical AI researchers are quick to point out – as Rosenblatt already noted – that neural net modelers have so far had difficulty dealing with step-wise problem solving. Connectionists respond that they are confident that they will solve that problem in time. This response, however, reminds one too much of the way that the symbol manipulators in the sixties responded to the criticism that their programs were poor at the perception of patterns. The old struggle between intellectualists who, because they can do context-free logic think they have a handle on everyday cognition but are poor at understanding perception, and gestaltists who have the rudiments of an account of perception[67] but none of everyday cognition, goes on. One might think, using the metaphor of the right and left brain, that perhaps the brain/mind uses each strategy when appropriate. The problem would then be how to combine them. One cannot just switch back and forth for, as Heidegger and the gestaltists saw, the pragmatic background plays a crucial role in determining relevance even in everyday logic and problem solving, and experts in any field, even logic, grasp operations in terms of their functional similarities.

It is even premature to consider combining the two approaches, since so far neither has accomplished enough to be on solid ground. Neural network modeling may simply be getting a deserved chance to fail as did the symbolic approach.

Still there is an important difference to remember as each research program struggles on. The physical symbol system approach seems to be failing because it is simply false to assume that there must be a theory of every domain. Neural network modeling, however, is not committed to this or any other philosophical assumption. However, simply building an interactive net sufficiently similar to the one our brain has evolved may be just too hard. Indeed, the common sense knowledge problem, which has blocked the progress of symbolic representation techniques for fifteen years, may be looming on the neural net horizon, although connectionists may not yet recognize it. All neural net modelers agree that for a net to be intelligent it must be able to generalize, that is, given sufficient examples of inputs associated with one particular output, it should associate further inputs of the same type with that same output. The questions arises, however: What counts as the same type? The designer of the net has a specific definition in mind of the type required for a reasonable generalization, and counts it a success if the net generalizes to other instances of this type. But when the net produces an unexpected association can one say it has failed to generalize? One could equally well say that the net has all along been acting on a different definition of the type in question and that that difference has just been revealed. (All the "continue this sequence" questions found on intelligence tests really have more than one possible answer but most humans share a sense of what is simple and reasonable and therefore acceptable.)

Neural network modelers attempt to avoid this ambiguity and make the net produce "reasonable" generalizations by considering only a pre-specified allowable family of generalizations, i.e., allowable transformations which will count as acceptable generalizations (the hypothesis space). They then attempt to design the architecture of their nets so that the net transforms inputs into outputs only in ways which are in the hypothesis space. Generalization will then be possible only on the designer's terms. While a few examples will be insufficient to identify uniquely the appropriate member of the hypothesis space, after enough examples only one hypothesis will account for all the examples. The net will then have learned the appropriate generalization principle, i.e., all further input will produce what, from the designer's point of view, is the appropriate output.

The problem here is that the designer has determined by means of the architecture of the net that certain possible generalizations will never be found. All this is well and good for toy problems in which there is no question of what constitutes a reasonable generalization, but in real-world situa-

[67] For a recent influential account of perception that denies the need for mental representation see, James J. Gibson, *The Ecological Approach to Visual Perception*, (Boston: Houghton Mifflin Company, 1979). Gibson and Rosenblatt collaborated on a research paper for the Air Force in 1955.

tions a large part of human intelligence consists in generalizing in ways appropriate to the context. If the designer restricts the net to a pre-defined class of appropriate responses, the net will be exhibiting the intelligence built into it by the designer for that context but will not have the common sense that would enable it to adapt to other contexts as would a truly human intelligence.

Perhaps a net must share size, architecture and initial connection configuration with the human brain if it is to share our sense of appropriate generalizations. If it is to learn from its own "experiences" to make associations that are human-like rather than be taught to make associations which have been specified by its trainer, it must also share our sense of appropriateness of outputs, and this means it must share our needs, desires, and emotions and have a human-like body with the same physical movements, abilities and possible injuries.

If Heidegger and Wittgenstein are right, human beings are much more holistic than neural nets. Intelligence has to be motivated by purposes in the organism and other goals picked up by the organism from an on-going culture. If the minimum unit of analysis is that of a whole organism geared into a whole cultural world, neural nets as well as symbolically programmed computers, still have a very long way to go.

References

[1] M. Boden, *Artificial Intelligence and Natural Man*, (New York: Basic Books, 1977).

[2] H. Dreyfus, *What Computers Can't Do*, (New York: Harper and Row, 2nd edition, 1979).

[3] H. Dreyfus, *Being-in-the-world: A Commentary on Division I of Being and Time*, (Cambridge: MIT Press/Bradford Books, 1988).

[4] H. Dreyfus ed., *Husserl, Intentionality and Cognitive Science*, (Cambridge: MIT Press/Bradford Books, 1982).

[5] Hubert and Stuart Dreyfus, *Mind Over Machine*, (New York: Free Press/Macmillan, 1986).

[6] James J. Gibson, *The Ecological Approach to Visual Perception*, (Boston: Houghton Mifflin Company, 1979).

[7] J. Haugeland, *Artificial Intelligence: The Very Idea*, (Cambridge: Bradford/MIT Press, 1985).

[8] Martin Heidegger, *Being and Time*, (New York: Harper and Row), 1962, Sections 14-21,

[9] Geoffrey Hinton, "Learning Distributed Representations of Concep," *Proceedings of the 8th Annual Conference Cognitive Science Society*, Amherst, Mass., Aug. 1986

[10] Hobbes, *Leviathan*, (New York: Library of Liberal Arts, 1958), p. 45.

[11] E. Husserl, *Ideas Pertaining to a Pure Phenomenology and to a Phenomenological Philosophy*, trans. F. Kersten, (The Hague: Nijhoff, 1982).

[12] Edmund Husserl, *Crisis of European Sciences and Transcendental Phenomenology*, trans. D. Carr, (Evanston: Northwestern University Press, 1970), p. 149.

[13] Husserl, *Ideen zu einer reinen Phanomenologie und phenomenologischen Philosophie*, Drittes Buch, 1913, #75, p. 134.

[14] Edmund Husserl, *Ideen Zu Einer Reinen Ph:anomenologie und Ph:anomenologischen Philosophie*, Nijhoff, 1950.

[15] Husserl, *Cartesians Meditations*, pp. 54-55.

[16] Husserl, *Formal and Transcendental Logic*, trans. D. Cairns (The Hague: Nijhoff, 1969), p. 244.

[17] Edmund Husserl, *Cartesians Meditations*, trans. D. Cairns, (The Hague: Nijhoff, 1960) p. 45

[18] Gina Kolata, "How Can Computers Get Common Sense?", *Science*, Vol. 217, 24 September 1982. p. 1237.

[19] Imre Lakatos, *Philosophical Papers*, ed. J. Worrall, (Cambridge: Cambridge University Press, 1978).

[20] Leibniz, *Selections*, ed. Philip Wiener (New York: Scribner, 1951), p. 18.

[21] Sir James Lighthill, "Artificial Intelligence: A General Survey" in *Artificial Intelligence: a paper symposium*, (London: Science Research Council, 1973).

[22] Pamela McCorduck, *Machines Who Think*, (San Francisco: W.H. Freeman and Company, 1979), p. 87.

[23] Marvin Minsky, "A Framework for Representing Knowledge," in *Mind Design*, J. Haugeland, ed., p. 96.

[24] Minsky, *Computation: Finite and Infinite Machines*, (New York: Prentice Hall, 1977), p. 2.

[25] Marvin Minsky and Seymour Papert, *Perceptrons: An Introduction to Computational Geometry*, (Cambridge: The MIT Press, 1969), p. 19.

[26] Allen Newell, "Intellectual Issues in the History of Artificial Intelligence", in *The Study of Information: Interdisciplinary Messages*, F. Machlup and U. Mansfield, eds. (New York: John Wiley and Sons, 1983), p. 196.

[27] Allen Newell and Herbert Simon, "Computer Sciencê as Empirical Inquiry: Symbols and Search", reprinted in *Mind Design*, John Haugeland, ed., (Cambridge: Bradford/MIT Press, 1981), p. 41.

[28] Frank Rosenblatt, "Strategic Approaches to the Study of Brain Models," *Principles of Self-Organization*, H. von Foerster, ed., (Pergamon Press, 1962), p. 386.

[29] F. Rosenblatt, *Principles of Neurodynamics, Perceptrons and the Theory of Brain Mechanisms*, (Washington, D.C.: Spartan Book, 1962), p. 292.

[30] F. Rosenblatt, *Mechanisation of Thought Processes: Proceedings of a Symposium held at the National Physical Laboratory, November 1958. Vol. 1.*, p. 449., (London: HM Stationery Office).

[31] David Rumelhart and James McClelland: *Parallel Distributed Processing*,

[32] Rumelhart and Norman, "A Comparison of Models," *Parallel Models of Associative Memory*, Hinton and Anderson eds., Lawrence Erlbaum Associates, Publishers, 1981, p. 3.

[33] H. Simon, *The Shape of Automation for Men and Management*, Harper and Row, 1965, p. 96.

[34] Herbert Simon and Allen Newell, "Heuristic Problem Solving: The Next Advance in Operations Research", *Operations Research*, Vol. 6 (January- February 1958), p. 6.

[35] Paul Smolensky, "On the proper treatment of connectionism", *Behavioral and Brain Sciences*, final draft, p. 1, summer 1987.

[36] T. Winograd, "Artificial Intelligence and Language Comprehension," in *Artificial Intelligence and Language Comprehension*, National Institute of Education, 1976, p. 9.

[37] T. Winograd, "Computer Software for Working with Language," *Scientific American*, September 1984, p. 142.

[38] Wittgenstein, *Philosophical Investigations*, (Oxford: Basil Blackwell, 1953).

[39] Ludwig Wittgenstein, *Philosophical Remarks*, University of Chicago Press, 1975.

[40] L. Wittgenstein, *Last Writings on the Philosophy of Psychology, Vol. I*, Chicago University Press, 1982, #504, p. 66e. (Translation corrected).

[41] L. Wittgenstein, *Tractatus Logico-Philosophicus*, (London: Routledge and Kegan Paul, 1960).

Thinking Machines: Can There Be? Are We?

Terry Winograd
Stanford University, Computer Science Dept., Stanford, CA 95305-2140, USA
E-mail: winograd@cs.stanford.edu

Keywords: thinking machines, broader understanding

Edited by: Matjaž Gams
Received: October 18, 1994 **Revised:** September 28, 1995 **Accepted:** October 25, 1995

Artificial intelligence researchers predict that "thinking machines" will take over our mental work, just as their mechanical predecessors were intended to eliminate physical drudgery. Critics have argued with equal fervor that "thinking machine" is a contradiction in terms. Computers, with their foundations of cold logic, can never be creative or insightful or possess real judgment. Although my own understanding developed through active participation in artificial intelligence research, I have now come to recognize a larger grain of truth in the criticisms than in the enthusiastic predictions. The source of the difficulties will not be found in the details of silicon micro-circuits or of Boolean logic, but in a basic philosophy of patchwork rationalism that has guided the research. In this paper I review the guiding principles of artificial intelligence and argue that as now conceived it is limited to a very particular kind of intelligence: one that can usefully be likened to bureaucracy. In conclusion I will briefly introduce an orientation I call hermeneutic constructivism and illustrate how it can lead to an alternative path of design.

1 Introduction

Futurologists have proclaimed the birth of a new species, machina sapiens, that will share (perhaps usurp) our place as the intelligent sovereigns of our earthly domain. These "thinking machines" will take over our burdensome mental chores, just as their mechanical predecessors were intended to eliminate physical drudgery. Eventually they will apply their "ultra-intelligence" to solving all of our problems. Any thoughts of resisting this inevitable evolution is just a form of "speciesism," born from a romantic and irrational attachment to the peculiarities of the human organism.

Critics have argued with equal fervor that "thinking machine" is an oxymoron – a contradiction in terms. Computers, with their foundations of cold logic, can never be creative or insightful or possess real judgment. No matter how competent they appear, they do not have the genuine intentionality that is at the heart of human understanding. The vain pretensions of those who seek to understand mind as computation can be dismissed as yet another demonstration of the arrogance of modern science.

Although my own understanding developed through active participation in artificial intelligence research, I have now come to recognize a larger grain of truth in the criticisms than in the enthusiastic predictions[1]. But the story is more complex. The issues need not (perhaps cannot) be debated as fundamental questions concerning the place of humanity in the universe. Indeed, artificial intelligence has not achieved creativity, insight and judgment. But its shortcomings are far more mundane: we have not yet been able to

[1]The work presented here was supported by the System Development Foundation under a grant to the Center for the Study of Language and Information at Stanford University. A version of this paper was presented at the conference on "Humans, Animals, and Machines: Boundaries and Projections," sponsored by the Stanford Humanities Center in April, 1987. This paper was published as Winograd, Terry, "Thinking machines: Can there be? Are We?," in James Sheehan and Morton Sosna, eds., The Boundaries of Humanity: Humans, Animals, Machines, Berkeley: University of California Press, 1991, pp. 198-223. Reprinted with permission.

construct a machine with even a modicum of common sense or one that can converse on everyday topics in ordinary language.

The source of the difficulties will not be found in the details of silicon micro-circuits or of Boolean logic. The basic philosophy that has guided the research is shallow and inadequate, and has not received sufficient scrutiny. It is drawn from the traditions of rationalism and logical empiricism but has taken a novel turn away from its predecessors. This new "patchwork rationalism" will be our subject of examination.

First, we will review the guiding principles of artificial intelligence and see how they are embodied in current research. Then we will look at the fruits of that research. I will argue that "artificial intelligence" as now conceived is limited to a very particular kind of intelligence: one that can usefully be likened to bureaucracy in its rigidity, obtuseness, and inability to adapt to changing circumstances. The weakness comes not from insufficient development of the technology, but from the inadequacy of the basic tenets.

But, as with bureaucracy, weaknesses go hand in hand with unique strengths. Through a reinterpretation and re-formulation of the techniques that have been developed, we can anticipate and design appropriate and valuable uses. In conclusion I will briefly introduce an orientation I call hermeneutic constructivism and illustrate how it can lead down this alternative path of design.

2 The mechanization of rationality

In their quest for mechanical explanations of (or substitutes for) human reason, researchers in artificial intelligence are heirs to a long tradition. In his "Discourse on the method of properly guiding the reason in the search of truth in the sciences" (1637), Descartes initiated the quest for a systematic method of rationality. Although Descartes himself did not believe that reason could be achieved through mechanical devices, his understanding laid the groundwork for the symbol-processing machines of the modern age.

In 1651, Hobbes described reason as symbolic calculation:

"When a man reasoneth, he does nothing else but conceive a sum total, from addition of parcels; or conceive a remainder... These operations are not incident to numbers only, but to all manner of things that can be added together, and taken one out of another... the logicians teach the same in consequences of words; adding together two names to make an affirmation, and two affirmations to make a syllogism; and many syllogisms to make a demonstration."[2]

Leibniz (as described by Russell)

"... cherished through his life the hope of discovering a kind of generalized mathematics, which he called Characteristica Universalis, by means of which thinking could be replaced by calculation. "If we had it," he says "we should be able to reason in metaphysics and morals in much the same way as in geometry and analysis. If controversies were to arise, there would be no more need of disputation between two philosophers than between two accountants. For it would suffice to take their pencils in their hands, to sit down to their slates, and to say to each other... 'Let us calculate.' "[3]

Behind this program of mechanical reason was a faith in a rational and ultimately understandable universe. The model of "Let us calculate" is that of Euclidean geometry, in which a small set of clear and self- evident postulates provides a basis for generating the right answers (given sufficient diligence) to the most complex and vexing problems. Reasonable men could be relied upon to agree on the postulates and the methods, and therefore dispute could only arise from mistaken calculation.

The empiricists turned to physical experience and experiment as the true basis of knowledge. But in rejecting the a priori status of the propositions on which reasoning was based, they did not abandon the vision of rigorous (potentially mechanizable) logical procedures. For our purposes here, it will suffice to adopt a broader characterization, in which much of both rationalism and empiricism fall within a common "rationalistic tradition."[4] This label subsumes the varied (and at times hotly opposed) inheritors of Descartes' legacy — those who seek to achieve rational reason through a precise method of symbolic cal-

[2]Hobbes, Leviathan, quoted in Haugeland, Artificial Intelligence: The Very Idea, p24.

[3]Russell, A History of Western Philosophy, p. 592.

[4]See Chapter 2 of Winograd and Flores, Understanding Computers and Cognition.

culation.

The electronic computer gave new embodiment to mechanical rationality, making it possible to derive the consequences of precisely specified rules, even when huge amounts of calculation are required. The first decades of computing emphasized the application of numerical techniques. Researchers in operations research and decision theory addressed policy questions by developing complex mathematical models of social and political systems and calculating the results of proposed alternatives.[5] Although these techniques work well in specialized cases (such as scheduling delivery vehicles or controlling the operations in a refinery), they proved inadequate for the broader problems to which they were applied. The "mathematization" of experience required simplifications that made the computer results – accurate as they might be with respect to the models – meaningless in the world.

Although there are still attempts to quantify matters of social import (for example in applying mathematical risk analysis to decisions about nuclear power), there is an overall disillusionment with the potential for adequately reducing human concerns to a precise set of numbers and equations.[6] The developers of artificial intelligence have rejected traditional mathematical modelling in favor of an emphasis on symbolic – rather than numerical – formalisms. Leibniz's "Let us calculate" is taken in Hobbes broader sense to include not just numbers but also "affirmations" and "syllogisms."

3 The promise of artificial intelligence

Attempts to duplicate formal non-numerical reasoning on a machine date back to the earliest computers, but the endeavor began in earnest with the artificial intelligence (AI) projects of the mid 1950s.[7] The goals were ambitious: to fully duplicate the human capacities of thought and language on a digital computer. Early claims that a complete theory of intelligence would be achieved within a few decades have long since been abandoned, but the reach has not diminished. For example, a recent book by Minsky (one of the founders of AI) offers computational models for phenonemena as diverse as conflict, pain and pleasure, the self, the soul, consciousness, confusion, genius, infant emotion, foreign accents, and freedom of will.[8]

In building models of mind, there are two distinct but complementary goals. On the one hand is the quest to explain human mental processes as thoroughly and unambiguously as physics explains the functioning of ordinary mechanical devices. On the other hand is the drive to create intelligent tools — machines that apply intelligence to serve some purpose, regardless of how closely they mimic the details of human intelligence. At times these two enterprises have gone hand in hand, at others they have led down separate paths.

Researchers such as Newell and Simon (two other founding fathers of artificial intelligence) have sought precise and scientifically testable theories of more modest scope than Minsky suggests. In reducing the study of mind to the formulation of rule-governed operations on symbol systems, they focus on detailed aspects of cognitive functioning, using empirical measures such as memory capacity and reaction time. They hypothesize specific "mental architectures" and compare their detailed performance with human experimental results.[9] It is difficult to measure the success of this enterprise. The tasks that have been examined (such as puzzle-solving and the ability to remember abbreviations for computer commands) do not even begin to approach a representative sample of human cognitive abilities, for reasons we will examine below.

On the other side lies the goal of practical system building. In the late 1970s, the field of artificial intelligence was drastically affected by the continuing precipitous drop in computing costs. Techniques that previously demanded highly specialized and costly equipment came within the reach of commercial users. A new term, "knowledge

[5]One large-scale and quite controversial example was the MIT/Club of Rome simulation of the world social and economic future (The Limits of Growth).

[6]See, for example, the discussions in Davis and Hersh, Descartes' Dream.

[7]See Gardner, The Mind's New Science, for an overview of the historical context.

[8]These are among the section headings in Minsky, The Society of Mind.

[9]See, for example, Newell & Simon, Human Problem Solving, and Laird et al., Universal Subgoaling and Chunking.

FACTS:
Tank #23 contains sulfuric acid.
The plaintiff was injured by a portable power saw.
RULES:
If the sulfate ion test is positive, the spill material is sulfuric acid.
If the plaintiff was negligent in the use of the product, the theory of contributory negligence applies.

Figure 1: Rules for an expert system (from D. Waterman, A Guide to Expert Systems, p. 16).

engineering," was coined to indicate a shift to the pragmatic interests of the engineer, rather than the scientist's search for theoretical knowledge.

"Expert systems," as the new programs were called, incorporate "knowledge bases" made up of simple facts and "if... then" rules, as illustrated in Figure 1.

These systems do not attempt to explain human intelligence in detail, but are justified in terms of their practical applications, for which extravagant claims have been made.

Humans need expert systems, but the problem is they don't often believe it... At least one high-performance medical diagnosis program sits unused because the physicians it was designed to assist didn't perceive that they needed such assistance; they were wrong, but that doesn't matter... There's a manifest destiny in information processing, in knowledge systems, a continent we shall all spread out upon sooner or later.[10]

The high hopes and ambitious aspirations of knowledge engineering are well documented, and the claims are often taken at face value, even in serious intellectual discussions. In fact, although a few widely-known systems illustrate specific potentials, the successes are still isolated pinnacles in a landscape of research prototypes, feasibility studies, and preliminary versions. It is difficult to get a clear picture of what has been accomplished and to make a realistic assessment of what is yet to come. We need to begin by examining the difficulties with the fundamental methods these programs employ.

[10]Feigenbaum and McCorduck, pp. 86, 95, 152.

4 The foundations of artificial intelligence

Artificial intelligence draws its appeal from the same ideas of mechanized reasoning that attracted Descartes, Leibniz and Hobbes, but it differs from the more classical forms of rationalism in a critical way. Descartes wanted his method to stand on a bedrock of clear and self-evident truths. Logical empiricism sought truth through observation and the refinement of formal theories that predicted experimental results. Artificial intelligence has abandoned the quest for certainty and truth. The new patchwork rationalism is built upon mounds of "micro-truths" gleaned through common sense introspection, ad hoc programming and so-called "knowledge acquisition" techniques for interviewing experts. The grounding on this shifting sand is pragmatic in the crude sense – "If it seems to be working, it's right."

The resulting patchwork defies logic. Minsky observes:

"For generations, scientists and philosophers have tried to explain ordinary reasoning in terms of logical principles – with virtually no success. I suspect this enterprise failed because it was looking in the wrong direction: common sense works so well not because it is an approximation of logic; logic is only a small part of our great accumulation of different, useful ways to chain things together."[11]

In the days before computing, "ways to chain things together" would have remained a vague metaphor. But the computer can perform arbitrary symbol manipulations that we interpret as having logical import. It is easy to build a program to which we enter "Most birds can fly" and "Tweety is a bird" and which then produces "Tweety can fly" according to a regular (although logically questionable) rule. The artificial intelligence methodology does not demand a logically correct answer, but one that works sufficiently

[11]Minsky, The Society of Mind, p. 187. Although Minsky's view is prevalent among AI researchers, not all of his colleagues agree that thought is so open-endedly non-logical. McCarthy (co-founder with Minsky of the MIT AI- lab), for example, is exploring new forms of logic that attempt to preserve the rigor of ordinary deduction, while dealing with some of the properties of commonsense reasoning, as described in the papers in Bobrow (ed.), Special Issue on Nonmonotonic Logic.

often to be "heuristically adequate."

In a way, this approach is very attractive. Everyday human thought does not follow the rigid strictures of formal deduction. Perhaps we can devise some more flexible (and even fallible) system that operates according to mechanical principles, but more accurately mirrors the mind.

But this appeal is subtly deceptive. Minsky places the blame for lack of success in explaining ordinary reasoning on the rigidity of logic, and does not raise the more fundamental questions about the nature of all symbolic representations and of formal (though possibly "non-logical") systems of rules for manipulating them. There are basic limits to what can be done with symbol manipulation, regardless of how many "different, useful ways to chain things together" one invents. The reduction of mind to the interactive sum of decontextualized fragments is ultimately impossible and misleading. But before elaborating on the problems, let us first review some assumptions on which this work proceeds:

1. Intelligence is exhibited by "physical symbol systems."
2. These systems carry out symbol manipulations that correspond to some kind of "problem solving."
3. Intelligence is embodied as a large collection of fragments of "knowledge."

4.1 The physical symbol system hypothesis

The fundamental principle is the identification of intelligence with the functioning of a rule-governed symbol-manipulating device. It has been most explicitly stated by Newell and Simon:

"A physical symbol system has the necessary and sufficient means for general intelligent action,... By 'general intelligent action' we wish to indicate the same scope of intelligence we see in human action: that in any real situation behavior appropriate to the ends of the system and adaptive to the demands of the environment can occur, within some limits of speed and complexity."[12]

This "physical symbol system hypothesis" presupposes materialism: the claim that all of the observed properties of intelligent beings can ultimately be explained in terms of lawful physical processes. It adds the claim that these processes can be described at a level of abstraction in which all relevant aspects of physical state can be understood as the encoding of symbol structures and that the activities can be adequately characterized as systematic application of symbol manipulation rules.

The essential link is representation — the encoding of the relevant aspects of the world. Newell lays this out explicitly:

"An intelligent agent is embedded in a task environment; a task statement enters via a perceptual component and is encoded in an initial representation. Whence starts a cycle of activity in which a recognition occurs... of a method to use to attempt the problem. The method draws upon a memory of general world knowledge... It is clear to us all what representation is in this picture. It is the data structures that hold the problem and will be processed into a form that makes the solution available. Additionally, it is the data structures that hold the world knowledge and will be processed to acquire parts of the solution or to obtain guidance in constructing it."[13] [emphasis in original].

Complete and systematic symbolic representation is crucial to the paradigm. The rules followed by the machine can deal only with the symbols, not their interpretation.

4.2 Problem-solving, inference and search

Newell and Simon's physical symbol systems aspire not to an idealized rationality, but to "behavior appropriate to the ends of the system and adaptive to the demands of the environment." This shift reflects the formulation that won Simon a Nobel prize in economics. He supplanted decision theories based on optimization with a theory of "satisficing" – effectively using finite decision – making resources to come up with adequate, but not necessarily optimal plans of action.

As artificial intelligence developed in the 1950s and 60s, this methodology was formalized in the techniques of "heuristic search."

The task that a symbol system is faced with,

[12]Newell & Simon, Computer science as emiprical inquiry (their speech accepting the ACM Turing Award - the computer science equivalent of the Nobel Prize).

[13]Newell, The knowledge level, p. 88.

then, when it is presented with a problem and a problem space, is to use its limited processing resources to generate possible solutions, one after another, until it finds one that satisfies the problem-defining test.[14]

The "problem space" is a formal structure that can be thought of as enumerating the results of all possible sequences of actions that might be taken by the program. In a program for playing chess, for example, the problem space is generated by the possible sequences of moves. The number of possibilities grows exponentially with the number of moves, and is beyond practical reach after a small number. However, one can limit search in this space by following heuristics that operate on the basis of local cues ("If one of your pieces could be taken on the opponent's next move, try moving it..."). There have been a number of variations on this basic theme, all of which are based on explicit representations of the problem space and the heuristics for operating within it.

Figure 1 illustrated some rules and facts from expert systems. These are not represented in the computer as sentences in English, but as symbols intended to correspond to the natural language terms. As these examples indicate, the domains are naturally far richer and more complex than can be captured by such simple rules. A lawyer will have many questions about whether a plaintiff was 'negligent,' but for the program it is a simple matter of whether a certain symbolic expression of the form "Negligent(x)" appears in the store of representations, or whether there is a rule of the form "If... then Negligent(x)," whose conditions can be satisfied.

There has been a great deal of technical debate over the detailed form of rules, but two principles are taken for granted in essentially all of the work:

1. Each rule is true in a limited (situation-dependent), not absolute sense.
2. The overall result derives from the synergistic combination of rules, in a pattern that need not (in fact could not in general) be anticipated in writing them.

For example, there may be cases in which the "sulfate ion test is positive" even though the spill is not sulfuric acid. The overall architecture of the rule-manipulating system may lead to a conclusion being drawn that violates one of these rules (on the basis of other rules). The question is not whether each of the rules is true, but whether the output of the program as a whole is "appropriate." The knowledge engineers hope that by devising and tuning such rules they can capture more than the deductive logic of the domain:

> While conventional programs deal with facts, expert systems handle 'lore'... the rules of thumb, the hunches, the intuition and capacity for judgement that are seldom explicitly laid down but which form the basis of an expert's skill, acquired over a lifetime's experience.[15]

This ad hoc nature of the logic applies equally to the cognitive models of Newell and Simon, in which a large collection of separate "production rules" operate on a symbolic store or "working memory." Each production rule specifies a step to be carried out on the symbols in the store, and the overall architecture determines which will be carried out in what order. The symbols don't stand for chemical spills and law, but for hypothesized psychological features, such as the symbolic contents of short term memory. Individual rules do things like moving an element to the front of the memory or erasing it. The cognitive modeler does not build an overall model of the system's performance on a task, but designs the individual rules in hopes that appropriate behavior will emerge from their interaction.

Minsky makes explicit this assumption that intelligence will emerge from computational interactions among a plethora of small pieces.

> I'll call 'Society of Mind' this scheme in which each mind is made of many smaller processes. These we'll call agents. Each mental agent by itself can only do some simple thing that needs no mind or thought at all. Yet when we join these agents in societies – in certain very special ways – this leads to true intelligence.[16]

Minsky's theory is quite different from Newell's cognitive architectures. In place of finely tuned clockworks of precise production rules we find an impressionistic pastiche of metaphors. Minsky illustrates his view in a simple 'micro-world' of toy blocks, populated by agents such as BUILDER (which stacks up the blocks), ADD (which adds a

[14]Newell & Simon, Computer science as empirical inquiry, p. 121.

[15]Michie and Johnston, The Creative Computer, p. 35.

[16]Minsky, The Society of Mind, p. 17.

single block to a stack), and the like:

For example, BUILDER's agents require no sense of meaning to do their work; ADD merely has to turn on GET and PUT. Then GET and PUT do not need any subtle sense of what those turn-on signals "mean" — because they're wired up to do only what they're wired up to do.[17]

These agents seem like simple computer subroutines — program fragments that perform a single well-defined task. But a subsequent chapter describes an interaction between the BUILDER agent and the WRECKER agent, which are parts of a PLAY-WITH-BLOCKS agent:

Inside an actual child, the agencies responsible for BUILDING and WRECKING might indeed become versatile enough to negotiate by offering support for one another's goals. "Please, WRECKER, wait a moment more till BUILDER adds just one more block: it's worth it for a louder crash!"[18]

With a simple "might indeed become versatile...", we have slipped from a technically feasible but limited notion of agents as subroutines, to an impressionistic description of a society of homunculi, conversing with each other in ordinary language. This sleight of hand is at the center of the theory. It takes an almost childish leap of faith to assume that the modes of explanation that work for the details of block manipulation will be adequate for understanding conflict, consciousness, genius, and freedom of will.

One cannot dismiss this as an isolated fantasy. Minsky is one of the major figures in artificial intelligence and he is only stating in a simplistic form a view that permeates the field. In looking at the development of computer technology, one cannot help but be struck by the successes at reducing complex and varied tasks to systematic combinations of elementary operations. Why not, then, make the jump to the mind. If we are no more than protoplasm-based physical symbol systems, the reduction must be possible and only our current lack of knowledge prevents us from explicating it in detail, all the way from BUILDER's clever ploy down to the logical circuitry.

[17]Ibid., p. 67.

[18]Ibid., p. 33.

4.3 Knowledge as a commodity

All of the approaches described above depend on interactions among large numbers of individual elements: rules, productions, or agents. No one of these elements can be taken as representing a substantial understandable truth, but this doesn't matter since somehow the conglomeration will come out all right. But how can we have any confidence that it will? The proposed answer is a typical one of our modern society: "More is better!" "Knowledge is power, and more knowledge is more power."

A widely-used expert systems text declares:

"It wasn't until the late 1970s that AI scientists began to realize something quite important: The problem-solving power of a program comes from the knowledge it possesses, not just from the formalisms and inference schemes it employees. The conceptual breakthrough was made and can be quite simply stated. To make a program intelligent, provide it with lots of high-quality, specific knowledge about some problem area."[19]

This statement is typical of much writing on expert systems, both in the parochial perspective that inflates a homily into a "conceptual breakthrough" and in its use of slogans like "high-quality knowledge." Michie (the Dean of artificial intelligence in Britain) predicts :

"[Expert systems]... can actually help to codify and improve expert human knowledge, taking what was fragmentary, inconsistent and error-infested and turning it into knowledge that is more precise, reliable and comprehensive. This new process, with its enormous potential for the future, we call 'knowledge refining.' "[20]

Feigenbaum proclaims:

"The miracle product is knowledge, and the Japanese are planning to package and sell it the way other nations package and sell energy, food, or manufactured goods... The essence of the computer revolution is that the burden of producing the future knowledge of the world will be transferred from human heads to machine artifacts."[21]

Knowledge is a kind of commodity — to be produced, refined, and packaged. The knowledge en-

[19]Waterman, A Guide to Expert Systems, p. 4 [emphasis in the original].

[20]Michie and Johnston, The Creative Computer, p. 129.

[21]Feigenbaum & McCorduck, The Fifth Generation, pp. 12, 40.

gineers are not concerned with the age-old epistemological problems of what constitutes knowledge or understanding. They are hard at work on techniques of "knowledge acquisition" and see it as just a matter of sufficient money and effort:

We have the opportunity at this moment to do a new version of Diderot's Encyclopedia, a gathering up of all knowledge – not just the academic kind, but the informal, experiential, heuristic kind – to be fused, amplified, and distributed, all at orders of magnitude difference in cost, speed, volume, and usefulness over what we have now.[22]

Lenat has embarked on this task of "encod[ing] all the world's knowledge down to some level of detail." The plan projects an initial entry of about 400 articles from a desk encyclopedia (leading to 10,000 paragraphs worth of material), followed by hiring a large number of "knowledge enterers" to add "the last 99 percent." There is little concern that foundational problems might get in the way. Lenat asserts that "AI has for many years understood enough about representation and inference to tackle this project, but no one has sat down and done it."[23]

5 The fundamental problems

The optimistic claims for artificial intelligence have far outstripped the achievements, both in the theoretical enterprise of cognitive modelling and in the practical application of expert systems.

Cognitive models seek experimental fit with measured human behavior but the enterprise is fraught with methodological difficulty, as it straddles the wide chasm between the engineering bravado of computer science and the careful empiricism of experimental psychology. When a computer program duplicates to some degree some carefully restricted aspect of human behavior, what have we learned? It is all too easy to write a program that would produce that particular behavior, and all too hard to build one that covers a sufficiently general range to inspire confidence. As Pylyshyn (an enthusiastic participant in cognitive science) observes:

"Most current computational models of cognition are vastly underconstrained and ad hoc; they are contrivances assembled to mimic arbitrary pieces of behavior, with insufficient concern for explicating the principles in virtue of which such behavior is exhibited and with little regard for a precise understanding."[24]

Newell and his colleagues' painstaking attention to detailed architecture of production systems is an attempt to better constrain the computational model, in hopes that experiments can then test detailed hypotheses. As with much of experimental psychology, a highly artificial experimental situation is required to get results that can be sensibly interpreted at all. Proponents argue that the methods and theoretical foundations that are being applied to micro-behavior will eventually be extended and generalized to cover the full range of cognitive phenomena. As with Minsky, this leap from the micro-structure to the whole human is one of faith.

In the case of expert systems, there is a more immediate concern. Applied AI is widely seen as a means of managing processes that have grown too complex or too rapid for unassisted humans. Major industrial and governmental organizations are mounting serious efforts to build expert systems for tasks such as air traffic control, nuclear power plant operation and – most distressingly – the control of weapons systems. These projects are justified with claims of generality and flexibility for AI programs. They ignore or downplay the difficulties that will make the programs almost certain to fail in just those cases where their success is most critical.

It is a commonplace in the field to describe expert systems as "brittle"-able to operate only within a narrow range of situations. The problem here is not just one of insufficient engineering, but is a direct consequence of the nature of rule-based systems. We will examine three manifestations of the problem: gaps of anticipation; blindness of representation; and restriction of the domain.

5.1 Gaps of anticipation

In creating a program or knowledge base, one takes into account as many factors and connections as feasible. But in any realistically complex domain, this gives at best a spotty coverage. The person designing a system for dealing with acid spills may not consider the possibility of rain le-

[22]Ibid., p. 229 [emphasis in the original].

[23]Lenat, CYC, p. 75.

[24]Pylyshyn, Computation and Cognition, p. xv.

aking into the building, or of a power failure, or that a labelled bottle does not contain what it purports to. A human expert faced with a problem in such a circumstance falls back on common sense and a general background of knowledge.

The hope of patchwork rationalism is that with a sufficiently large body of rules, the thought-through spots will successfully interpolate to the wastelands in between. Having written rule A with one circumstance in mind and rule B with another, the two rules in combination will succeed in yet a third. This strategy is the justification for the claim that AI systems are more flexible than conventional programs. There is a grain of truth in the comparison, but it is deceptive. The program applies the rules blindly with erratic results. In many cases, the price of flexibility (the ability to operate in combinations of contingencies not considered by the programmer) is irreparable and inscrutable failure.

In attempting to overcome this brittleness, expert systems are built with many thousands of rules, trying to cover all of the relevant situations and to provide representations for all potentially relevant aspects of context. One system for medical diagnosis, called CADUCEUS (originally INTERNIST) has 500 disease profiles, 350 disease variations, several thousand symptoms, and 6,500 rules describing relations among symptoms. After fifteen years of development, the system is still not on the market. According to one report, it gave a correct diagnosis in only 75% of its carefully selected test cases. Nevertheless, Myers, the medical expert who developed it, "believes that the addition of another 50 [diseases] will make the system workable and, more importantly, practical."[25]

Human experts develop their skills through observing and acting in many thousands of cases. AI researchers argue that this results in their remembering a huge repertoire of specialized "patterns" (complex symbolic rules) that allow them to discriminate situations with expert finesse and to recognize appropriate actions. But it is far from obvious whether the result of experience can be adequately formalized as a repertoire of discrete patterns.[26] To say that "all of the world's knowledge" could be explicitly articulated in any symbolic form (computational or not) we must assume the possibility of reducing all forms of tacit knowledge (skills, intuition, and the like) to explicit facts and rules. Heidegger and other phenomenologists have challenged this, and many of the strongest criticisms of artificial intelligence are based on the phenomenological analysis of human understanding as a "readiness-to-hand" of action in the world, rather than as the manipulation of "present-to-hand" representations.[27]

Be that as it may, it is clear that the corresponding task in building expert systems is extremely difficult, if not theoretically impossible. The knowledge engineer attempts to provide the program with rules that correspond to the expert's experience. The rules are modified through analyzing examples in which the original rules break down. But the patchwork nature of the rules makes this extremely difficult. Failure in a particular case may not be attributable to a particular rule, but rather to a chance combination of rules that are in other circumstances quite useful. The breakdown may not even provide sharp criteria for knowing what to change, as with a chess program that is just failing to come up with good moves. The problem here is not simply one of scale or computational complexity. Computers are perfectly capable of operating on millions of elements. The problem is one of human understanding — the ability of a person to understand how a new situation experienced in the world is related to an existing set of representations, and to possible modifications of those representations.

In trying to remove the potentially unreliable "human element," expert systems conceal it. The power plant will no longer fail because a reactor-operator falls asleep, but because a knowledge engineer didn't think of putting in a rule specifying how to handle a particular failure when the emergency system is undergoing its periodic test, and the backup system is out of order. No amount of refinement and articulation can guarantee the absence of such breakdowns. The hope that a system based on patchwork rationalism will respond "appropriately" in such cases is just that: a hope, and one that can engender dangerous illusions of safety and security.

[25]Newquist, The machinery of medical diagnosis, p. 70.

[26]See the discussion in H. Dreyfus and S. Dreyfus, Mind Over Machine.

[27]See, for example, H. Dreyfus, What Computers Can't Do, and Winograd & Flores, Understanding Computers and Cognition.

5.2 The blindness of representation

The second problem lies in the symbol system hypothesis itself. In order to characterize a situation in symbolic form, one uses a system of basic distinctions, or terms. Rules deal with the interrelations among the terms, not with their interpretations in the world.

Consider ordinary words as an analogy. Imagine that a doctor asks a nurse "Is the patient eating?" If they are deciding whether to perform an examination, the request might be paraphrased "Is she eating at this moment?" If the patient is in the hospital for anorexia and the doctor is checking the efficacy of the treatment, it might be more like "Has the patient eaten some minimal amount in the past day?" If the patient has recently undergone surgery, it might mean "Has the patient taken any nutrition by mouth," and so on. In responding, a person interprets the sentence as having relevance in the current situation, and will typically respond appropriately without conscious choosing among meanings.

In order to build a successful symbol system, decontextualized meaning is necessary — terms must be stripped of open-ended ambiguities and shadings. A medical expert system might have a rule of the form: "IF Eating(x) THEN...," which is to be applied only if the patient is eating, along with others of the form "IF ... THEN Eating (x)" which determine when that condition holds. Unless everyone who writes or reads a rule interprets the primitive term "Eating" in the same way, the rules have no consistent interpretation and the results are unpredictable.

In response to this, one can try to refine the vocabulary. "Currently-Dining" and "Taking-Solids" could replace the more generic term, or we could add construal rules, such as "in a context of immediate action, take 'Eating' to mean 'Currently-Dining'." Such approaches work for the cases that programmers anticipate, but of course are subject to the infinite regress of trying to decontextualize context. The new terms or rules themselves depend on interpretation that is not represented in the system.

5.3 Restriction of the domain

A consequence of decontextualized representation is the difficulty of creating AI programs in any but the most carefully restricted domains, where almost all of the knowledge required to perform the task is special to that domain (i.e., little common sense knowledge is required). One can find specialized tasks for which appropriate limitations can be achieved, but these do not include the majority of work in commerce, medicine, law, or the other professions demanding expertise.

Holt characterized the situation:

"A brilliant chess move while the room is filling with smoke because the house is burning down does not show intelligence. If the capacity for brilliant chess moves without regard to life circumstances deserves a name, I would naturally call it 'artificial intelligence.' "[28]

The brilliance of a move is with respect to a well-defined domain: the rules of chess. But acting as an expert doctor, attorney, or engineer takes the other kind of intelligence: knowing what makes sense in a situation. The most successful artificial intelligence programs have operated in the detached puzzle-like domains of board games and technical analysis, not those demanding understanding of human lives, motivations, and social interaction. Attempts to cross into these difficult territories, such as a program said to "understand tales involving friendship and adultery,"[29] proceed by replacing the real situation with a cartoon-like caricature, governed by simplistic rules whose inadequacy is immediately obvious (even to the creators, who argue that they simply need further elaboration).

This reformulation of a domain to a narrower, more precise one can lead to systems that give correct answers to irrelevant problems. This is of concern not only when actions are based directly on the output of the computer system (as in one controlling weapons systems), but also when, for example, medical expert systems are used to evaluate the work of physicians.[30] Since the system is based on a reduced representation of the situation, it systematically (if invisibly) values some aspects of care while remaining blind to others. Doctors whose salaries, promotions, or accredi-

[28] Holt, Remarks made at ARPA Principal Investigators' Conference, p. 1.

[29] See the discussion of the BORIS program in Winograd and Flores, Understanding Computers and Cognition, pp. 121ff.

[30] See Athanasiou, High-tech politic, The case of artificial intelligence, p. 24.

tation depend on the review of their actions by such a program will find their practice being subtly shaped to its mold.

The attempt to encode "the world's knowledge" inevitably leads to this kind of simplification. Every explicit representation of knowledge bears within it a background of cultural orientation that does not appear as explicit claims, but is manifest in the very terms in which the 'facts' are expressed and in the judgment of what constitutes a fact. An encyclopedia is not a compendium of "refined knowledge," but a statement within a tradition and a culture. By calling an electronic encyclopedia a 'knowledge base' we mystify its source and its grounding in a tradition and background.

6 The bureaucracy of mind

Many observers have noted the natural affinity between computers and bureaucracy. Lee argues that "bureaucracies are the most ubiquitous form of artificial intelligence... Just as scientific management found its idealization in automation and programmable production robots, one might consider an artificiallyntelligent knowledge-based system as the ideal bureaucrat..."[31] Lee's stated goal is "improved bureaucratic software engineering," but his analogy suggests more.

Stated simply, the techniques of artificial intelligence are to the mind what bureaucracy is to human social interaction.

In today's popular discussion, bureaucracy is seen as an evil—a pathology of large organizations and repressive governments. But in his classic work on bureaucracy, Weber argued its great advantages over earlier, less formalized systems, calling it the "unambiguous yardstick for the modernization of the state." He notes that "bureaucracy has a 'rational' character, with rules, means-ends calculus, and matter-of-factness predominating,"[32] and that it succeeds in "eliminating from official business love, hatred, and all purely personal, irrational, and emotional elements which escape calculation."[33]

The decisive reason for the advance of bureaucratic organization has always been its purely technical superiority over any other form of organization. The fully developed bureaucratic apparatus compares with other organizations exactly as does the machine with the non-mechanical modes of production. Precision, speed, unambiguity, knowledge of the files, continuity, discretion, unity, strict subordination, reduction of friction and of material and personal costs – these are raised to the optimum point in the strictly bureaucratic administration.[34]

The benefits of bureaucracy follow from the reduction of judgment to the systematic application of explicitly articulated rules. Bureaucracy achieves a predictability and manageability that is missing in earlier forms of organization. There are striking similarities here with the arguments given for the benefits of expert systems, and equally striking analogies with the shortcomings as pointed out, for example, by March and Simon:

"The reduction in personalized relationships, the increased internalization of rules, and the decreased search for alternatives combine to make the behavior of members of the organization highly predictable; i.e., they result in an increase in the rigidity of behavior of participants [which] increases the amount of difficulty with clients of the organization and complicates the achievement of client satisftion."[35]

Given Simon's role in artificial intelligence, it ironic that he notes these weaknesses of human-embodied rule systems, but sees the behavior of rule-based physical symbol systems as "adaptive to the demands of the environment." Indeed, systems based on symbol manipulation exhibit the rigidities of bureaucracies, and are most problematic in dealing with "client satisfaction" — the mismatch between the decontextualized application of rules and the human interpretation of the symbols that appear in them. Bureaucracy is most successful in a world that is stable and repetitive — where the rules can be followed without interpretive judgments. Expert systems are best in just the same situations. Their successes have been in stable and precise technical areas, where exceptions are not the rule.

Michie's claim that expert systems can encode "the rules of thumb, the hunches, the intuition and capacity for judgement..." is wrong in the

[31] Lee, Bureaucracy as artificial intelligence, p. 127.
[32] Weber, Economy and Society, p. 1002.
[33] Ibid., p. 975.
[34] Ibid., p. 973 [emphasis in original].
[35] March and Simon, Organizations, p. 38 [emphasis in original].

same way that it is wrong to seek a full account of an organization in its formal rules and procedures. Modern sociologists have gone beyond Weber's analysis, pointing to the informal organization and tacit knowledge that make organizations work effectively. This closely parallels the importance of tacit knowledge in individual expertise. Without it we get rigidity and occasional but irreparable failure.

The depersonalization of knowledge in expert systems also has obvious parallels with bureaucracy. When a person views his or her job as the correct application of a set of rules (whether human-invoked or computer-based), there is a loss of personal responsibility or commitment. The "I just follow the rules" of the bureaucratic clerk has its direct analog in "That's what the knowledge base says." The individual is not committed to appropriate results (as judged in some larger human context), but to faithful application of the procedures.

This forgetfulness of individual commitment is perhaps the most subtle and dangerous consequence of patchwork rationality. The person who puts rules into a knowledge base cannot be committed to the consequences of applying them in a situation he or she cannot foresee. The person who applies them cannot be committed to their formulation or to the mechanics by which they produce an answer. The result belongs to no one. When we speak here of "commitment," we mean something more general than the kind of accountability that is argued in court. There is a deep sense in which every use of language is a reflection of commitment, as we will see in the following section.

7 Alternatives

We began with the question of thinking machines-devices that mechanically reproduce human capacities of thought and language. We have seen how this question has been reformulated in the pursuit of artificial intelligence, to reflect a particular design based on patchwork rationalism. We have argued that the current direction will be inadequate to explain or construct real intelligence.

But, one might ask, does that mean that no machine could exhibit intelligence? Is artificial intelligence inherently impossible, or is it just fiendishly difficult? To answer sensibly we must first ask what we mean by "machine." There is a simple a priori proof that machines can be intelligent if we accept that our own brains are (in Minsky's provocative words) nothing but "meat machines." If we take "machine" to stand for any physically constituted device subject to the causal laws of nature, then the question reduces to one of materialism, and is not to be resolved through computer research. If, on the other hand, we take machine to mean "physical symbol system" then there is ground for a strong skepticism. This skepticism has become visible among practitioners of artificial intelligence as well as the critics.

7.1 Emergent intelligence

The innovative ideas of cybernetics a few decades ago led to two contrasting research programmes. One, which we have examined here, took the course of symbol processing. The other was based on modelling neural activity and led to the work on "perceptrons," a research line that was discounted for many years as fruitless and is now being rehabilitated in "connectionist" theories, based on "massively parallel distributed processing." In this work, each computing element (analogous to a neuron) operates on simple general principles, and intelligence emerges from the evolving patterns of interaction.[36]

Connectionism is one manifestation of what Turkle calls "emergent AI."[37] The fundamental intuition guiding this work is that cognitive structure in organisms emerges through learning and experience, not through explicit representation and programming. The problems of blindness and domain limitation described above need not apply to a system that has developed through situated experience.

It is not yet clear whether we will see a turn back towards the heritage of cybernetics or simply a "massively parallel" variant of current cognitive theory and symbol processing design. Although the new connectionism may breathe new life into

[36]For a historical account and analysis of the current debates, see H. Dreyfus, Making a mind vs. modeling the brain. For a technical view, see Rumelhart and MacLelland, Parallel Distributed Processing. Maturana and Varela, in The Tree of Knowledge, offer a broad philosophy of cognition on this base.

[37]Turkle, A new romantic reaction.

cognitive modelling research, it suffers an uneasy balance between symbolic and physiological description. Its spirit harks back to the cybernetic concern with real biological systems, but the detailed models typically assume a simplistic representational base much closer to traditional artificial intelligence. Connectionism, like its parent cognitive theory, must be placed in the category of brash unproved hypotheses, which have not really begun to deal with the complexities of mind, and whose current explanatory power is extremely limited.

In one of the earliest critiques of artificial intelligence, Dreyfus compared it to alchemy.[38] Seekers after the glitter of intelligence are misguided in trying to cast it from the base metal of computing. There is an amusing epilogue to this analogy: in fact, the alchemists were right. Lead can be converted into gold by a particle accelerator hurling appropriate beams at lead targets. The AI visionaries may be right in the same way, and they are likely to be wrong in the same way. There is no reason but hubris to believe that we are any closer to understanding intelligence than the alchemists were to the secrets of nuclear physics. The ability to create a glistening simulacrum should not fool us into thinking the rest is "just a matter of encoding a sufficient part of the world's knowledge" or into a quest for the philosopher's stone of "massively parallel processing."

7.2 Hermeneutic constructivism

Discussions of the problems and dangers of computers often leave the impression that on the whole we would be better off if we could return to the pre-computer era. In a similar vein one might decry the advent of written language, which created many new problems. For example, Weber attributes the emergence of bureaucracy to the spread of writing and literacy, which made it possible to create and maintain systems of rules. Indeed, the written word made bureaucracy possible, but that is far from a full account of its relevance to human society.

The computer is a physical embodiment of the symbolic calculations envisaged by Hobbes and Leibniz. As such, it is really not a thinking machine, but a language machine. The very notion of "symbol system" is inherently linguistic and what we duplicate in our programs with their rules and propositions is really a form of verbal argument, not the workings of mind. It is tempting – but ultimately misleading – to project the image of rational discourse (and its reflection in conscious introspection) onto the design of embodied intelligence. In taking inner discourse as a model for the activity of Minsky's tiny agents, or of productions that determine what token to process next, artificial intelligence has operated with the faith that mind is linguistic down to the microscopic level.

But the utility of the technology need not depend on this faith. The computer, like writing, is fundamentally a communication medium-one that is unique in its ability to perform complex manipulations on the linguistic objects it stores and transmits. We can reinterpret the technology of artificial intelligence in a new background, with new consequences. In doing so we draw on an alternative philosophical grounding, which I will call hermeneutic constructivism.

We begin with some fundamental questions about what language is and how it works. In this we draw on work in hermeneutics (the study of interpretation) and phenomenology, as developed by Heidegger and Gadamer, along with the concepts of language action developed from the later works of Wittgenstein through the speech act philosophy of Austin, Searle, and Habermas.[39]

Two guiding principles emerge: People create their world through language. Language is always interpreted in a tacitly understood background.

Austin pointed out that "performative" sentences do not convey information about the world, but act to change that world. "You're hired," when uttered in appropriate conditions, creates — not describes — a situation of employment. Searle applied this insight to mundane language actions such as asking questions and agreeing to do something. Habermas extended it further, showing how sentences we would naively consider statements of fact have force by virtue of an act of commitment by the speaker.

The essential presupposition for the success of [a language] act consists in the speaker's entering into a specific engagement, so that the hearer can

[38] H. Dreyfus, Alchemy and artificial intelligence.

[39] See Chapter 5 of Winograd & Flores, Understanding Computers and Cognition, for an overview.

References

[1] Athanasiou, Tom, High-tech politics: The case of artificial intelligence, Socialist Review (1987), 7-35.

[2] Austin, J.L., How to Do Things with Words, Cambridge, Mass.: Harvard University Press, 1962.

[3] Bobrow, Daniel (ed.), Special Issue on Nonmonotonic Logic, Artificial Intelligence, 13:1 (Jan 1980).

[4] Club of Rome, The Limits to Growth, New York: Universe Books, 1972.

[5] Davis, Philip J. and Reuben Hersh, Descartes' Dream: The World According to Mathematics, San Diego: Harcourt Brace, 1986.

[6] Dreyfus, Hubert, Alchemy and artificial intelligence, Rand Corporation Paper P-3244, December 1965.

[7] Dreyfus, Hubert, What Computers Can't Do: A Critique of Artificial Reason, New York: Harper and Row, 1972 (2nd Edition with new Preface, 1979).

[8] Dreyfus, Hubert L., and Stuart E. Dreyfus, Making a mind vs. modeling the brain: AI back at the crossroads, (in this issue).

[9] Dreyfus, Hubert L., and Stuart E. Dreyfus, Mind Over Machine: The Power of Human Intuition and Expertise in the Era of the Computer, New York: Macmillan/The Free Press, 1985.

[10] Feigenbaum, Edward A., and Pamela McCorduck, The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World, Reading, Mass.: Addison-Wesley, 1983.

[11] Flores, C. Fernando, Management and Communication in the Office of the Future, Doctoral dissertation, University of California, Berkeley, 1982.

[12] Gardner, Howard, The Mind's New Science: A History of the Cognitive Revolution, New York: Basic Books, 1985.

[13] Habermas, Juergen, Communication and the Evolution of Society (translated by Thomas McCarthy), Boston: Beacon Press, 1979.

[14] Haugeland, John, Mind Design, Cambridge, Mass.: Bradford/MIT, 1981.

[15] Haugeland, John, Artificial Intelligence: The Very Idea. Cambridge, Mass.: Bradford/MIT, 1985.

[16] Holt, Anatol, Remarks made at ARPA Principal Investigators' Conference, Los Angeles, February 6-8, 1974 (unpublished manuscript).

[17] Howard, Robert, Systems design and social responsibility: The political implications of 'computer-supported cooperative work,' address delivered at the First Annual Conference on Computer-Supported Cooperative Work, Austin, Texas, December 1986.

[18] Laird, John, Paul Rosenbloom and Allen Newell, Universal Subgoaling and Chunking: The Automatic Generation and Learning of Goal Hierarchies, Hingham, Mass.: Kluwer, 1986.

[19] Lee, Ronald M., Bureaucracy as artificial intelligence, in L.B. Methlie and R.H. Sprague (eds.), Knowledge Representation for Decision Support Systems, New York: Elsevier (North-Holland), 1985, 125-132.

[20] Lee, Ronald M., Automating red tape: the performative vs. informative roles of bureaucratic documents, Office: Technology and People, 2 (1984), 187-204.

[21] Lenat, Douglas, CYC: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks, AI Magazine 6:4 (1986), 65-85.

[22] March, James G. and Herbert A. Simon, Organizations, New York: Wiley, 1958.

[23] Maturana, Humberto R. and Francisco Varela, The Tree of Knowledge, Boston: Shambhala, in press.

[24] Michie, Donald, and Rory Johnston, The Creative Computer, New York: Viking, 1984

[25] Minsky, Marvin, The Society of Mind, New York: Simon and Schuster, 1986.

[26] Newell, Allen, The knowledge level, Artificial Intelligence 18 (1982), 87-127.

[27] Newell, Allen and Herbert Simon, Computer science as empirical inquiry: Symbols and search, Communications of the ACM, 19 (March, 1976), 113-126. Reprinted in J. Haugeland (ed.), Mind Design, 35-66.

[28] Newquist, Harvey P. III, The machinery of medical diagnosis, AI Expert 2:5 (May, 1987), 69-71.

[29] Pylyshyn, Zenon, Computation and Cognition: Toward a Foundation for Cognitive Science, Cambridge, Mass.: Bradford/MIT, 1984.

[30] Roszak, Theodore, The Cult of Information: The Folklore of Computers and the True Art of Thinking, New York: Pantheon, 1986.

[31] Rumelhart, David, and James MacLelland, Parallel Distributed Processing: Explorations in the Microstructures of Cognition (2 volumes), Cambridge, Mass.: Bradford/MIT, 1986.

[32] Russell, Bertrand, A History of Western Philosophy, New York: Simon and Schuster, 1952.

[33] Simon, Herbert, Models of Thought, New Haven: Yale Univ. Press, 1979.

[34] Turkle, Sherry, A new romantic reaction: the computer as precipitant of anti-mechanistic definitions of the human, paper given at conference on Humans, Animals, Machines: Boundaries and Projections, Stanford University, April 1987.

[35] Waterman, Donald, A Guide to Expert Systems, Reading, Mass.: Addison- Wesley, 1986.

[36] Weber, Max, Economy and Society: An Outline of Interpretive Sociology, Berkeley: Univ. of California Press, 1968.

[37] Winograd, Terry, A language/action perspective on the design of cooperative work, Human-Computer Interaction, 1987 (in press).

[38] Winograd, Terry and Fernando Flores, Understanding Computers and Cognition: A New Foundation for Design, Norwood New Jersey: Ablex, 1986.

"Strong AI": an Adolescent Disorder

Donald Michie
Professor Emeritus, University of Edinburgh, UK
Associate Member, Josef Stefan Institute, Ljubljana, Slovenia

Keywords: strong and weak AI, Turing's test, middle-ground

Edited by: Matjaž Gams
Received: October 24, 1994 **Revised:** October 19, 1995 **Accepted:** November 4, 1995

Philosophers have distinguished two attitudes to the mechanization of thought. "Strong AI" says that given a sufficiency of well chosen axioms and deduction procedures we have all we need to program computers to out-think humans. "Weak AI" says that humans don't think in logical deductions anyway. So why not instead devote ourselves to (1) neural nets, or (2) ultra-parallelism, or (3) other ways of dispensing with symbolic domain-models?

"Weak AI" thus has diverse strands, united in a common objection to "strong AI", and articulated in popular writings, see for example Hubert Dreyfus (1979), John Searle (1990) and Roger Penrose (1989). How should one assess their objection?

1 Turing's Test and Postulates

If asked to investigate the alleged insolvency of the Fireproof Coal Corporation, a careful auditor first looks for evidence that such a corporation actually exists. Not being personally acquainted with adherents of the described "strong AI" school among professional colleagues, I looked for "strong AI" in the literature. I concluded that the description sufficiently matched an identifiable AI sub-community that flourished in the USA during the subject's adolescence (roughly 1965–1985) and probably retains professional adherents there today. Certainly the mind-set lives on in textbooks used for teaching. Because it steps backwards from Turing's original prescriptions, I use the label T-minus (for "Turing-minus") for this sub-school of symbolic AI. Misconceptions about T-minus may explain the philosophical attacks on "strong AI". A particularly salient misconception, fostered in some textbooks, is that T-minus traces intellectual paternity to Alan Turing's (1950) paper in which he proposed a test to settle whether a given machine could think. The machine must fool a remote interrogator into mistaking it for a human. In reality T-minus, while retaining the Test itself, implicitly rejects the postulate that accompanied it, namely that the role of machine learning is central, and necessary for attainment of the desired capability.

1.1 Intelligence is in the discourse, not the action

The capabilities that we call "intelligence" and "thought" are manifested not so much in problem solving as in discourse . In the context of Turing's imitation game, accurate problem solving was secondary. "It is claimed", he writes, "that the interrogator could distinguish the machine from the man simply by setting them a number of problems in arithmetic. The machine would be unmasked because of its deadly accuracy. The reply to this is simple. The machine (programmed for playing the game) would not attempt to give the right answers to the arithmetical problems. It would deliberately introduce mistakes in a manner calculated to confuse the interrogator."

Of course there may be machine intelligence in deciphering the arithmetical question, in invoking a suitable low-level solving routine, and in concocting sufficient hesitancy or error to make the response look human-like. But Turing does not present the arithmetical calculation itself as a manifestation of intelligence and thus avoids identifying intelligence with competence. The question

of whether intelligence would be of any use in a creature lacking a competent problem-solving system is a separate issue. But the exercise of even very great competence in an intellectual domain is not in itself proof of intelligence. Numerous computer triumphs of today, not restricted to arithmetic, remind us of this.

In confining "intelligence" to the discourse-testable functions of understanding and after-the-event reporting, Turing made a wise move. Those who failed to follow his example look foolish every time that an intelligent Grandmaster is defeated by a super-competent chess machine. Today's game-playing machines are profoundly deficient in understanding even of the games that they win, as witnessed by their inability to annotate them. Writing such commentaries (as chess masters commonly do) would require, precisely, intelligence, in the sense in which Turing understood the term.

1.2 Insufficiency of hand-crafting methods

Calculation shows hand-crafting to be infeasible for loading into the system the huge quantities of organized knowledge required for human-level intelligence. To estimate a lower bound, Turing made the optimistic assumption that a thousand megabits of program space might be sufficient for satisfactory playing of the imitation game, at least in the restricted form of play against a blind person, thus excluding from the accountancy the resource-hungry processes of visual perception. He continued: "At my present rate of working I produce about a thousand digits of program a day, so that about sixty workers, working steadily through the fifty years might accomplish the job, if nothing went into the wastepaper basket. Some more expeditious method seems desirable."

The fantasy dubbed "Strong AI" by its critics is blind (at least in American textbook expositions) not only to these early calculations of Turing's but also to the arithmetic of modern commercial programming. According to the most recent estimate known to me, a typical rate for a large system is 10 lines of installed code per programmer per day.

1.3 Need for mechanized learning and teachability

Having rejected direct programming of knowledge, and unaided deduction from programmed axioms, Turing turned to the bulk acquisition of knowledge through mechanized learning. He introduced the idea as follows.

> In the process of trying to imitate an adult human mind we are bound to think a good deal about the process which has brought it to the state that it is in. We may notice three components,
>
> 1. The initial state of the mind, say at birth,
> 2. The education to which it has been subjected,
> 3. Other experience, not to be described as education, to which it has been subjected.
>
> Instead of trying to produce a program to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain.

2 Definition and Difficulties of T-Minus

T-minus essentially re-instates a proposal advanced by Leibniz in the seventeenth century, namely that we could obtain definitive knowledge about the world by the sole means of algebraic and deductive manipulations of symbols applied to symbolically coded facts. Two only of Turing's many extensions to Leibniz' programme were retained, namely use of high-speed computing to perform the manipulations, and the test for detecting intelligent thought in the resulting system. Hence we could speak of "Leibniz plus" but have preferred "Turing minus", meaning Turing minus the central role of machine learning. We must now consider the persistence into the post-Turing era of this retrogressive position.

The following definition of AI is from an authoritative exponent.

> Artificial Intelligence is the enterprise of constructing a Physical Symbol System that can reliably pass the Turing Test. (M Ginsberg, 1993, Chapter 1).

Ginsberg's use here of "Physical Symbol System" follows Neweland Simon (1976), and is broad enough to cover any physical embodiment of a Universal Turing Machine. The definition is oriented towards engineering rather than philosophical goals, and Ginsberg emphasizes that "AI is fundamentally an engineering discipline, since our fundamental goal is that of building something." Moreover Ginsberg, like the Physical Symbol System Hypothesis' authors, speaks solely of the machine's reasoning from *facts and laws explicitly communicated to it*. Catastrophically, he excludes the nine tenths that in humans lies submerged below consciousness yet forms an essential core for run-time problem solving (see for example Michie 1993a, 1993b, 1994, 1995a, 1995b for reviews). It is interesting to contrast this hard-line aversion to the findings of psychology and brain science with the explicit distinction made in McCarthy's (1959) Advice Taker paper: "One might conjecture that division in man between conscious and unconscious thought occurs at the boundary between stimulus-response heuristics which do not have to be reasoned about but only obeyed, and the others which have to serve as premises in deduction."

At the time when McCarthy wrote those words, no means were known for acquiring the submerged tacit procedures from human brains for machine use (but see Shapiro (1987) for a later exercise in doing just this; see also Urbancic and Bratko (1994) for a review of "behavioural cloning" of control skills). But he was sufficiently aware of the massive dependency of high-level cognition on low-level tacit procedures that he saw their incorporation in the infrastructure of intelligent systems as a necessity. In this respect among others, McCarthy had moved forward from Turing, and can justly be seen as the intellectual forerunner of the "Turing-plus", or T-plus, doctrine that we shall later consider. One should, however, mention Turing's 1947 report as showing that he was not unaware of these tacit procedures and of their importance: "By long experience we can pick up and apply the most complicated rules without being able to enunciate them at all".

Before Ginsberg's book was written, T-minus was already being subjected to the standard validity test faced by any engineering doctrine: can you build it, and will it then stand? Indeed T-minus has so far been the only one of symbolic AI's construction doctrines to inspire serious attempts at all-round machine knowledgeability and intelligence. Readers of Ginsberg's textbook are not however informed of these attempts nor of their disappointing outcomes. It is as though a text on bridge-building not only ignored established knowledge of wind-induced oscillations in exposed structures but omitted mention of the disasters that have resulted. Ferguson's (1993) "Engineering and the Mind's Eye" cites the British construction engineer Sir Alfred Pugsley on the subject of the collapse of the Tacoma Narrows suspension bridge in 1940. The major lesson was "the unwisdom of allowing a particular profession to become too inward looking and so screened from relevant knowledge growing up in other fields around it." Had the designers of the Tacoma Narrows Bridge known more of aerodynamics, Pugsley concluded, the collapse might have been averted.

With the substitution of neuroscience for aerodynamics, relevant knowledge from which T-minus's disciples show signs of being screened includes evidence from cognitive and brain studies of solving problems, as opposed to justifying, documenting and explaining the solutions. The solving part is not generally performed symbolically, but through spatio-visual and above all unconscious intuitive processes (McCarthy's "stimulus-response heuristics"). Associated brain centres are anatomically remote from the cortical areas specialized for logical reasoning and language (see for example Squire, 1987). As to visuo-spatial thinking in engineering problem-solving, Ferguson (loc. cit.) supplies much relevant material. Examples abound in other works, many cited by Ferguson, on visual and intuitive components of problem-solving.

In the light of all this, rejection by T-minus of Turing's machine learning prescription seems blind indeed. Not only must the hand-crafting task, even on optimistic assumptions, take too long to accomplish. Worse, perhaps much of it is not susceptible to hand-crafting at all. This second possibility follows from the constant finding referred to earlier that expert problem solving depends critically on subcognitive skills inaccessible to conscious introspection. Yet unless algorithmic work-arounds can be devised in every case,

introspection is left as the only source on which hand-crafting of mental skills can draw. Later we will consider the inductivist sub-school of symbolic AI, fast becoming the leading edge of T-plus, which accepts the importance of subarticulate mental processes and dispenses almost entirely with introspective sources for accessing them. Instead, T-plus builds executable models of subarticulate skills by another route, that is by inductive learning from imitation of skilled behaviour (see Urbancic and Bratko, 1994, for the interesting case of control skills).

3 T-Minus Under Test

So how has it gone with "the enterprise of constructing a Physical Symbol System that can reliably pass the Turing Test"? Two substantial T-minus projects were launched within a few years of each other. Japan's Fifth Generation (5G) project (conducted between 1979–1981) was aimed at the declared goal of human-level intelligence by the end of the 1980's. The early history and divergent later course has been reviewed by Michie (1988). In 1984 a group led by Lenat at the Microelectronics and Computer Technology Corporation in Texas, USA, launched a ten-year project known as CYC. Its aim was to build a huge interactive knowledge base spanning most of what humans call common sense, that was eventually to "grow by assimilating textbooks, literature, newspapers, etc." Numerous large databases would also be accessible to the system. During the closing years, "a cadre of teachers" would replace hand-crafting.

More than ten years on, we may note that both projects missed their stated marks. Before its collapse, the Tacoma Narrows suspension bridge at least looked like a bridge and behaved as a bridge. As elsewhere analysed (Michie, 1994; Gams, 1995) neither of the above-mentioned T-minus projects ever attained even the semblance of human-level knowledge and intelligence. What faults, then, underlay these failures?

Neglect of Turing's child-machine postulate. 5G initially relied on hand-crafting. Realization then took hold in the project's leadership that inductive knowledge acquisition should be recognized as the central focus for the project. But at that stage the initial goals had been diffused by complexities of sponsorship from a diversity of private companies in addition to the MITI governmental agency. By the time that a productive impetus had developed for inductive logic programming and other learning methods, 5G had diverged from its initial performance specifications. The main effort became concentrated into what proved to be a successful programme of transfer into industry of existing techniques.

CYC, on the other hand, acknowledged the necessary role of inductive learning from the start, but hung back from its systematic development. It is not clear from Lenat and Guha's (1989) interim report how this came about.

Neglect of the multiplicity of "understanding". The word "intelligence" is derived from the Latin for "understanding" There is moreover agreement that to merit description as intelligent, a system's responses must, at the least, give the appearance of understanding the domain of discourse, that is to say, of utilizing a stored domain model. Some leading AI workers see the storage and use of only one kind of model of a domain as not going far enough. Minsky (1994) writes: "If you understand something in only one way, then you really do not understand it at all. The secret of what anything means to us depends on how we have connected it to all the other things we know. That is why, when someone learns 'by rote', we say that they do not really understand."

There is a duality in human concepts. They are undeniably and commonly used, just as Minsky proposes, to represent one and the same notion in different ways, for example in symbol-strings and in pictures, with frequent and fluent interconversions between representations. Thus, there are two ways of seeing that an equilateral triangle has equal base angles. One is by Euclidean proof. The other is by mentally rotating it round the perpendicular and observing that the flipped image fits the unflipped one.

Neglect of science. Ginsberg remarks that all good engineering rests on a scientific foundation, and contrasts the views of extremist technologists with those of mathematical philosophers working on AI's scientific foundations. There are, it seems, AI technologists who believe "that the scientific foundation of AI has already been laid, and that the work that remains is engineering in nature." Against this "are people who believe that AI has

many fundamental scientific problems still to be solved; that the goal of constructing an intelligent artifact today is not dissimilar to the goal of building a nuclear reactor in 1920 ..."

John McCarthy's school of research inaugurated by his 1959 "Programs with common sense" represents this second position. McCarthy has in particular pointed to a wealth of concepts that are fundamental to everyday discourse, concerned with temporal sequence, causality, intention, capability, context-dependence and other common usages. The latter may include such phenomena as the deployment by two or more interacting agents of models of each other. Formalizing these everyday notions has so far largely resisted the efforts of AI logicians. A good interim overview has been made available by Ginsberg (1987).

Returning to the imitation game, we may reasonably enquire as follows. AI still lacks machine-executable languages in which elementary day-to-day transactions and inferences of human life may be expressed. So long as the lack persists, how can any project of 5G or CYC type hope to endow an automated conversationalist with the skill of describing such transactions?

Neglect of user requirements. From the time of Archimedes through Leonardo's to the present day, engineering design has taken the client's *statement of requirement* as starting point. Turing's proposal of a machine for playing the imitation game departs from this. What customers were there for a disembodied general-purpose artificial intelligence? There was, and is, no shortage of general-purpose natural intelligences. They can be found in abundance on any street corner. Two circumstances need to be kept in mind.

(1) Turing's paper was published in a journal devoted to the philosophy of mind. He was as much concerned to drive home a philosophical point as to launch a potential industry. This I believe explains the Turing Test's curiously free-floating character. But the closing part of his paper, which few commentators appear to read, discusses implementation, including the question of where to start. Turing suggests more circumscribed domains such as game-playing and robotics, for both of which healthy commercial markets have since appeared.

(2) In so far as the paper addresses non-philosophical issues they are concerned with engineering science rather than technology. Turing's vision was of intellectual tasks designed to serve as laboratory tests linking the work of theoreticians and experimentalists. Work of this kind precedes market considerations, just as the years of experimentation by the Wright brothers preceded the era of military and commercial aircraft design. Turing's thinking is conveyed in a closing passage:

"We may hope that machines will eventually compete with men in all purely intellectual fields. But which are the best ones to start with? Even this is a difficult decision. Many people think that a very abstract activity, like the playing of chess, would be best. It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child. Things would be pointed out and named, etc. Again I do not know what the right answer is, but I think both approaches should be tried."

5G and CYC began in the 1980's, remote in time from Turing's teachable blank slate. With the rise of expert systems, marketable innovation at the technology end of the AI spectrum was already an established fact, and a wide range of knowledge-based and rule-learning software techniques and tools were available. Market-oriented specializations of Turing's general-purpose question-answerer would not have been discouraged by CYC's industrial sponsors. It is as though the Wright brothers had not been content with restricted objectives, and had insisted that their machine must be built to do everything that a bird can do, and in addition that it would do this without using wings (no use was made of industrial-strength learning tools).

Neglect not of one but of several factors lay at the root of of the failures of 5G and CYC (see also Gams, 1995, for analysis). The aim of the T-plus school of symbolic AI is to continue to develop these neglected factors.

4 Beyond the Turing Test

T-minus was evidently for a time sufficiently entrenched to be the source of design ideas for two major software engineering projects. Yet in spite of disappointing results from these attempted applications, T-minus still survives as a doc-

trine for instructing a new generation of AI engineers. This is quite evident from Ginsberg's book. But the action has already moved elsewhere.

An inductive sector of symbolic AI is becoming the main-stream approach to large-scale knowledge-acquisition and refinement. I refer to Machine Learning (ML), and in particular to recent extensions via Prolog and other Logic Programming formalisms. This trend is not something newly sprung to prominence in AI thinking. On the contrary, it is intrinsic in the ideas of the founders. We have already considered Turing's own position. It was endorsed and extended by symbolic AI's grand architect John McCarthy. In his 1959 "Programs with common sense" he writes: "Our ultimate objective is to make programs that learn from experience as effectively as humans do." He goes on to warn that "in order for a program to be capable of learning something it must first be capable of being told it."

McCarthy is here speaking not of blind stimulus-response skills, some of which do not need explicit representation languages to be machine-learnable (e.g. by neural nets). He has in mind concept learning. Obviously until a hypothesis language is available in which a given concept is expressible, that concept cannot be explicitly learned. For this reason the rise of logic languages such as Prolog, and of the craft of Inductive Logic Programming (ILP) in particular, has played an important role by extending the expressivity of ML's hypothesis languages (Muggleton 1991). Radical progress along the McCarthy-Muggleton line is now necessary before a successful CYC-type project can be envisaged.

Along with the critical issue of hypothesis languages, extensions of ILP are required. These include facilities for hierarchical structuring of domains into contexts, for incorporating object-oriented features, for interfacing with constraint satisfaction programming, for manufacture of new attributes by constructive induction, and for the seamless incorporation of capabilities of uncertain inference. In a recent review of some of ML's problems and current progress (Michie 1995b), I have stressed a further gap that still separates achievement from potential. Inductive Logic Programming packages, even after thousands of hours of significant theory-discovery in a given domain, end up no better at solving the next problem than at the start. Consider, for example, a human biomolecular chemist's inductive inferences concerning likely activities of newly synthesized drugs, as studied using ILP by Sternberg and colleagues (1994). These come faster as his or her experience grows of a given domain of compounds. So here is a kind of "meta-learning", crucial in human intelligence. Active brains somehow incrementally assimilate statistico-logical properties of learning environments into background knowledge in ways that AI has not yet attempted to emulate.

5 Concluding Remarks

The time has come to venture beyond the horizons of the Turing Test. The IT market of today is looking to computers for more than intelligent chat. The need is for specialized intelligences that can deploy and articulate mastery of knowledge-intensive domains in science, engineering, medicine, pharmaceuticals, and finance.

Advances in symbolic learning are gradually establishing a sufficient technical foundation for Turing's child-machine project. The year 2000 may see, not the first-base completion he had hoped for, but a belated start along the originally indicated line.

Meanwhile the "Strong AI" versus "Weak AI" debate, refuelled by Roger Penrose's 6-year-old book "The Emperor's New Mind", is again changing its character. Penrose (1994) has replaced the Strong/Weak dichotomy by a four-level gradation of attitudes. In his new book "Shadows of the Mind" these are distinguished by the symbols A, B, C and D (set in curly font). With the "Strong" and "Weak" dichotomy superseded, both sides of this debate may find that their artillery is being wasted on positions that are not so much untenable as abandoned. A middle-ground position, integrating (as I believe) useful features of the two extremes, can be found in a paper of mine on "Knowledge, learning and machine intelligence" (Michie, 1993a). The salient features of this "integrative school" are summarised in the last of the three items below.

Symbolic school. All thought can be modelled as deductive reasoning from logical descriptions of the world, and machine-processed in this form.

Neural school. Thought and knowledge are

mainly intuitive, non-introspectable, non-logical, associative, approximate, stochastic and "fuzzy". Fidelity to neurobiological fact demands that we build similar properties into AI software.

Integrative school. Thought requires co-operation between conscious reasoning, whether symbolic or visuo-spatial, and lower-level tacit operations. Different software representations are appropriate to different engineering requirements. The latter ordinarily cover not only run-time performance, but also self-documentation. Performance at high levels of domain complexity demands learning. Self-documentation of acquired knowledge demands that learning be symbolic.

Acknowledgement

This paper was completed while the author was in receipt of a Visiting Fellowship from the Engineering and Physical Sciences Research Council, UK (contract no. GR/J56806), at the Oxford University Computing Laboratory.

References

[1] Dreyfus, H. L. (1979). What Computers Can't Do: The Limits of Artificial Intelligence, New York: Harper & Row.

[2] Ferguson (1993). Engineering and the Mind's Eye, MIT Press.

[3] Gams, M. (1995). Strong vs. weak AI. Informatica, this issue.

[4] Ginsberg, M.L. (ed, 1987). Readings in Nonmonotonic Reasoning, Los Altos, CA: Morgan Kaufmann.

[5] Ginsberg, M.L. (1993) Essentials of Artificial Intelligence, San Francisco, CA: Morgan Kaufmann

[6] Lenat, D.B. and Guha, R.V. (1989). Building Large Knowledge-Based systems, Reading, MA: Addison-Wesley.

[7] McCarthy, J. (1959). Programs with common sense, In Mechanization of Thought Processes, 1, 77-84, London: Her Majesty's Stationery Office. Reprinted with additional material in Semantic information Processing, (ed. M. Minsky), Cambridge,MA and London, UK: The MIT Press, 1963.

[8] Michie, D. (1988). The fifth generation's unbridged gap, In A Half-Century of the Universal Turing Machine, (ed. R. Herken), Oxford: Oxford University Press.

[9] Michie, D. (1993a). Knowledge, learning and machine intelligence, Chapter 1 of Intelligent Systems , (ed. L. Sterling), New York: Plenum Press, pp. 1-19.

[10] Michie, D. (1993b). Turing's test and conscious thought, Artificial Intelligence, 60, 1-22.

[11] Michie, D. (1994). Consciousness as an engineering issue, Part 1, J. Consciousness Studies, 1 (2), 182-95.

[12] Michie, D. (1995a). Consciousness as an engineering issue, Part 2, J. Consciousness Studies, 2 (1), 52-66.

[13] Michie, D. (1995b). Problem decomposition and the learning of skills, In Machine Learning: ECML-95, ed. N. Lavrac and S. Wrobel, Lecture Notes in Artificial Intelligence, 914, Berlin, Heidelberg, New York: Springer Verlag, pp. 17-31.

[14] Minsky, M. (1994). Will robots inherit the earth? Scientific American, 271 (4), 86-91.

[15] Muggleton, S.H. (1991). Inductive Logic Programming. New Generation Computing, 8, 295-318.

[16] Newell, A. and Simon, H.A. (1976). Computer science as empirical enquiry, Communications of the ACM, 19, 35-66.

[17] Penrose, R. (1989). The Emperor's New Mind, Oxford, Oxford University Press.

[18] Penrose, R. (1994). Shadows of the Mind, Oxford: Oxford University Press.

[19] Searle, J.R. (1990). Is the brain's mind a computer program? Scientific American, 262, 20-25.

[20] Shapiro, A. (1987). Structured Induction in Expert Systems, Wokingham, UK; Reading, Menlo Park and NewYork, USA: Addison-Wesley.

[21] Squire, L.R. (1987), Memory and Brain, Oxford, Oxford University Press.

[22] Sternberg, M.J.E., King, R.D., Lewis, R.A. and Muggleton, S. (1994). Application of machine learning to structural molecular biology, Phil. Trans R. Soc. Lond. B, 344, 365-371.

[23] Turing, A.M. (1947). Intelligent machinery. Report submitted in 1948 to the National Physical Laboratory, UK. Published in Machine Intelligence 5, (ed. B.Meltzer and D.Michie), Edinburgh University Press, 1969.

[24] Turing, A.M. (1950). Computing machinery and intelligence, Mind, 59, 433-60.

[25] Urbancic, T. and Bratko, I. (1994). Reconstructing human skill with machine learning, In Proc. Europ. Conf. on AI (ECAI-94), Amsterdam.

Artificial Selfhood: The Path to True Artificial Intelligence

Ben Goertzel
Psychology Department, University of Western Australia
Nedlands WA 6009, Australia
E-mail: ben@psy.uwa.edu.au

Keywords: artificial intelligence, complex systems, self, psynet model

Edited by: Xindong Wu
Received: March 7, 1995 **Revised:** September 14, 1995 **Accepted:** October 9, 1995

In order to make strong AI a reality, formal logic and formal neural network theory must be abandoned in favor of complex systems science. The focus must be placed on large-scale emergent structures and dynamics. Creative intelligence is possible in a computer program, but only if the program is devised in such a way as to allow the spontaneous organization and emergence of "self- and reality-theories." In order to obtain such a program it may be necessary to program whole populations of interacting, "artificially intersubjective" AI programs.

1 Introduction

The march of progress toward true artificial intelligence has, in the opinion of many, come to a standstill. There has always been a tremendous gap between the creative adaptibility of natural intelligence and the impotent rigidity of existing AI programs. In the beginning, however, there was an underlying faith that this impotence and rigidity could be overcome. Today enthusiasm seems to be flagging. Very few AI researchers carry out research aimed explicitly at the goal of producing thinking computer programs. Instead the field of AI has been taken over by the specialized study of technical sub-problems. The original goal of the field of AI – producing computer programs displaying general intelligence – has been pushed off into the indefinite future.

We have sophisticated mathematical treatments which deal with one or two aspects of intelligence in isolation. We have "brittle" computer programs which operate effectively within their narrowly constrained domains. We have connectionist networks and genetic classifier systems which approach their narrow domains with slightly more flexibility, but require exquisite tuning, and still lack any ability to comprehend new types of situation. What we still do not have, however is a halfway decent understanding of what needs to be done in order to construct an intelligent computer program.

The goal of this paper is to suggest a simple answer for this "million dollar question." The principal ingredient needed to make strong AI a reality is, I claim, the *self*. A self is nothing mystical, it is a certain type of structure, evolving according to a certain type of dynamic, and depending on other structures and dynamics in specific ways. Self, I will argue, is necessary for creative adaptibility – for the spontaneous generation of new routines to deal with new situations. Current AI programs do not have selves, and, I will argue, they do not even have the component structures out of which selves are built. This is why they are so rigid and so impotent.

The fashioning of computer programs with selves – "artificial selfhood" – is not a theoretical impossibility, merely a difficult technical problem. For one thing, it clearly requires more memory and processing power than we currently have at our disposal. When sufficiently large MIMD parallel machines are developed, we will be able to make a serious attempt at writing an intelligent program. Until that time, it is foolish to expect success at strong AI. Even with appropriate hardware, however, serious difficulties may well arise, related to the problem of bringing a new self to maturity without a real "parent." It may perhaps

be necessary to resort to the evolution of populations of intelligences – what has been called AI through A-IS or "artificial intersubjectivity." But these difficulties cannot be confronted or fully understood until we have appropriate hardware. Arguments about the possibility of strong AI, based on the results of experimentation on 1995 computers, have more than a small taint of absurdity.

The plan of the remainder of the paper is as follows. Section 2 clarifies certain issues regarding the possibility of strong AI and the assumptions underlying different approaches to AI. Section 3 introduces the psychological notions of self- and reality-theories. Section 4 presents an argument for the crucial role of self- and reality-theories in creative intelligence. Section 5 outlines a mathematical model which uses ideas from complex systems science to explain the self-organization of self from simpler psychological constructs. Finally, Section 6 discusses A-IS or "artificial intersubjectivity," a possible technique for evolving AI systems with artificial selves.

2 Strong AI Is Possible

Before addressing the problems of AI, it is first necessary to establish what the problem of AI is *not*. It cannot be emphasized too strongly that there is no fundamental obstacle to the construction of intelligent computer programs. The argument is a simple and familiar one. First premise: humans are intelligent systems. Second premise: humans are also systems governed by the equations of physics. Third premise: the equations of physics can be approximated, to within any degree of accuracy, by space and time discrete iterations that can be represented as Turing machine programs. Conclusion: intelligent behavior can be simulated, to within any degree of accuracy, by Turing machine programs.

As I have pointed out in (Goertzel 1993), this argument can be made more rigorous by reference to the work of Deutsch (1985). Deutsch has defined a generalization of the deterministic Turing machine called "quantum computer," and he has proved that, according to the known principles of quantum physics, the quantum computer is capable of simulating any finite physical system to within any degree of accuracy. He has also proved that while a quantum computer can do everything an ordinary computer can, it cannot compute any functions besides those which an ordinary computer can run. However, quantum computers can compute some functions faster than Turing machines, in the average case sense and they have certain unique properties, such as the ability to generate truly random numbers.

Because of Deutsch's theorems, the assertion that brains can be modeled as *quantum* computers is not a vague hypothesis but a physical fact. One must still deal with the possibility that intelligent systems are fundamentally quantum systems, and cannot be accurately modeled by deterministic Turing machines. But there is no evidence that this is the case; the structures of the brain that are considered cognitively relevant (neurons, synapses, neurotransmitters, etc.) all operate on scales so large as to render quantum effects insignificant. This point is not universally agreed upon: Hameroff (1990) has argued for the cognitive relevance of the molecular structures in the cytoplasm, and (Goertzel 1995a) has argued for a relation between consciousness and true randomness. Finally, Penrose (1987) has argued that, not only are brains not classical systems, but they are not quantum systems either: they are systems that must be modeled using the equations of a yet-undiscovered theory of quantum gravity. But all these arguments in favor of the non-classical brain reside in the realm of speculation. It is a physical fact that the brain is a quantum computer, and hence deals only with computable functions. And, given the physical evidence, it is at this stage a very reasonable assumption that the brain is actually a deterministic computer.

This conclusion, however, is of limited practical utility. It leaves a very important question open: how to *find* these programs that carry out intelligent behaviors! We do not know the detailed structure of the human brain and body; and even if we did know it, the direct simulation of these systems on man-made machines might well be a very inefficient way to implement intelligence. The key question is, what are the properties that make humans intelligent?

The most pessimistic view is that only systems very, very similar to the human brain and body could ever be intelligent. At present this hypothesis cannot be proven or disproven. As has been pointed out, however, it is somewhat similar to

the proposition that only systems very, very similar to birds can fly. The difference is that, while we have recently learned how to build flying machines, we have not yet learned to build thinking machines.

On the other hand, it is possible that the key to intelligence lies in a certain collection of clever special-case problem-solving tools; or, perhaps in the possession of *any* sufficiently clever collection of special-case problem-solving tools. If this is the case then what AI researchers should be doing is to study small scale systems which are extremely effective at solving certain special problems. This, in fact, is what most AI researchers have been doing for the past few decades.

Finally, a third alternative is that the key to intelligence lies in certain *global structures*, certain overall patterns of organization. If this is the correct possibility, then the conclusion is that clever algorithms for solving toy problems are, while perhaps useful and even necessary, not the essence of intelligence. What matters most is the way that these clever algorithms are *organized.* This last point of view is the one adopted here. In particular, I wish to call attention to one particular "global structure," one particular overall pattern of organization: the self.

3 Self- and Reality-Theories

What is the self? Psychology provides this question with not one but many answers. One of the most AI-relevant answers, however, is that provided by Epstein's (1984) synthetic personality theory. Epstein argues that the self is a *theory.* This is a useful perspective for AI because theorization is something with which AI researchers have often been concerned.

Epstein's personality theory paints a refreshingly simple picture of the mind:

The human mind is so constituted that it tends to organize experience into conceptual systems. Human brains make connections between events, and, having made connections, they connect the connections, and so on, until they have developed an organized system of higher- and lower-order constructs that is both differentiated and integrated...

In addition to making connections between events, human brains have centers of pleasure and pain. The entire history of research on learning indicates that human and other higher-order animals are motivated to behave in a manner that brings pleasure and avoids pain. The human being thus has an interesting task cut out simply because of his or her biological structure: it is to construct a conceptual system in such a manner as to account for reality in a way that will produce the most favorable pleasure/pain ratio over the foreseeable future. This is obviously no simple matter, for the pursuit of pleasure and the acceptance of reality not infrequently appear to be at cross- purposes to each other.

He divides the human conceptual system into three categories: a self-theory, reality-theory, and connections between self-theory and reality-theory. And he notes that these theories may be judged by the same standards as theories in any other domain:

[Since] all individuals require theories in order to structure their experiences and to direct their lives, it follows that the adequacy of their adjustment can be determined by the adequacy of their theories. Like a theory in science, a personal theory of reality can be evaluated by the following attributes: extensivity [breadth or range], parsimony, empirical validity, internal consistency, testability and usefulness.

A person's self-theory consists of her best guesses about what kind of entity she is. In large part it consists of ideas about the relationship between herself and other things, or herself and other people. Some of these ideas may be wrong; but this is not the point. The point is that the theory as a whole must have the same qualities required of scientific theories. It must be able to explain familiar situations. It must be able to generate new explanations for unfamiliar situations. Its explanations must be detailed, sufficiently detailed to provide practical guidance for action. Insofar as possible, it should be concise and self-consistent.

The acquisition of a self-theory, in the development of the human mind, is intimately tied up with the body and the social network. The in-

fant must learn to distinguish her body from the remainder of the world. By systematically using the sense of touch – a sense which has never been reliably simulated in an AI program – she grows to understand the relation between herself and other things. Next, by watching other people she learns about people; inferring that she herself is a person, she learns about herself. She learns to guess what others are thinking about her, and then incorporates these opinions into her self-theory. Most crucially, a large part of a person's self-theory is also a *meta-self-theory*: a theory about how to acquire information for one's self-theory. For instance, an insecure person learns to adjust her self-theory by incorporating only negative information. A person continually thrust into novel situations learns to revise her self-theory rapidly and extensively based on the changing opinions of others – or else, perhaps, learns not to revise her self-theory based on the fickle evaluations of society. There is substantial evidence that a person's self- and reality-theories are directly related to their cognitive style; see for instance (Erdmann, 1988).

4 Self and Intelligence

My central thesis here is that the capacity for creative intelligence is dependent on the possession of effective self- and reality- theories. My argument for this point is not entirely an obvious one. I will argue that self- and reality- theories provide the *dynamic data structures* needed for flexible, adaptable, creative thought.

The single quality most lacking in current AI programs is the ability to go into a new situation and "get oriented." This is what is sometimes called the brittleness problem. Our AI programs, however intelligent in their specialized domains, do not know how to construct the representations that would allow them to apply their acumen to new situations. This general knack for "getting oriented" is something which humans acquire at a very early age.

People do not learn to get oriented all at once. They start out, as small children, by learning to orient themselves in relatively simple situations. By the time they build up to complicated social situations and abstract intellectual problems they have a good amount of experience behind them. Coming into a new situation, they are able to reason associatively: "What similar situations have I seen before?" And they are able to reason hierarchically: "What simpler situations is this one built out of?" By thus using the information gained from orienting themselves to previous situations, they are able to make reasonable guesses regarding the appropriate conceptual representations for the new situation. In other words, they build up a dynamic data structure consisting of new situations and the appropriate conceptual representations. This data structure is continually revised as new information that comes in, and it is used as a basis for acquiring new information. This data structure contains information about specific situation and also, more abstractly, about how to get oriented to new situations. My claim is that this data structure depends crucially on the self, so that it is not possible to learn how to get oriented to complex situations, without first having constructed complex self- and reality-theories.

In humans, self- and reality-theories are constructed in early childhood, as part of the process of getting oriented to simple, basic situations of human relationship – situations confronted by every human being by virtue of having a body and interacting with other humans. Thus, in the human mind, there are no given, a priori entities; everything bottoms out with the phenomenological and perceptual, with those very factors that play a central role in the initial formation of self- and reality-theories. Self- and reality- theories help us to build up these basic situations into more complex ones. They help us to define all the various parts of a complex system in terms of each other.

On the other hand, we provide our AI programs with concepts which "make no sense" to them, which they are intended to consider as given, a priori entities. They have no self- and reality-theories to help them build up these complex concepts out of simple experiential concepts – for, indeed, they have no body, no sense of sociality, and no simple experiential concepts. The perception/action/memory hierarchy bottoms out prematurely, there can be no functioning dynamic data structure for getting oriented, no creative adaptability, no true intelligence.

5 Self-Organization of the Self

This view of self and intelligence may seem overly vague and "hand-waving," in comparison to the rigorous theories proposed by logic-oriented AI researchers, and the intricate calculus-based proofs of neural network theorists. However, there is nothing inherently non-rigorous about the build-up of simpler theories and experiences into complex self- and reality-theories. It is perfectly possible to model this process mathematically; the mathematics involved is simply of a different sort from what one is used to seeing in AI. Instead of formal logic, one must make use of ideas from dynamical systems theory (Devaney 1988) and, more generally, the emerging science of complexity (Green and Bossomaier 1994). In this section I will briefly outline one way of mathematically modeling the self-organization of the self, based on the *psynet model* of (Goertzel 1993, 1993a, 1994, 1995, 1995a). The treatment here will necessarily be somewhat condensed; more extensive discussion may be found in the references.

The psynet model is based on the application of dynamical systems theory ideas to self-organizing agent systems (Agha 1988). An intelligent system is modeled as a collection of memory- and algorithm-carrying agents, which are able to act on other agents to produce yet other agents. Following (Goertzel 1994) these agents are called *magicians*. Cognitive structures are modeled as attractors of the magician-interaction dynamic. An hierarchy of nested attractor structures is postulated, culminating in the "dual network" of associative memory and hierarchical perception/control, and the "self- and reality-theory," a particular manifestation of the dual network.

Let S denote a set, to be called the space of magicians. Then S^*, the space of all finite sets composed of elements of S, with repeated elements allowed, is the space of *magician systems*. One may write

$$System_{t+1} = A(System_t) \tag{1}$$

where $System_t$ is an element of S^* denoting the magician population at time t, and A is the "action operator," a function mapping magician populations into magician populations.

Let us assume, for simplicity's sake, that all magician interactions are binary, i.e., involving one magician acting on another to create a third. In this instance the machinery of magician operations may be described by a binary algebraic operation $*$, so that where a, b and c are elements of S, $a * b = c$ is read "a acts on b to create c." The case of unary, ternary, etc. interactions may be treated in a similar way or, somewhat artificially, may be constructed as a corollary of the binary case.

The action operator may be decomposed as

$$A(X) = F(R(X)) \tag{2}$$

where R is the "raw potentiality" operator and F is a "filtering" operator. R is formally given by

$$\begin{aligned} R(System_t) = \\ R\left(\{a_1, a_2, \ldots, a_{n(t)}\}\right) \\ = \{a_i * a_j \| i, j = 1, \ldots, n(t)\} \end{aligned} \tag{3}$$

The purpose of R is to construct the "raw potentiality" of the magician system $System_t$, the set of all possible magician combinations which ensue from it. The role of the filtering operator F, on the other hand, is to select from the raw potentiality those combinations which are to be allowed to continue to the next time step. This selection may be all-or-none, or it may be probabilistic. To define the filtering operator formally, let P^* denote the a space of all probability distributions on the space magician systems S^*. Then, F is a function which maps $S^* \times S^*$ into P^*.

Magician systems, thus defined, are ageometric, or, to use the chemical term, "well-mixed." But one may also consider "graphical magician systems," magician systems that are specialized to some given graph G. Each magician is assigned a location on the graph as one of its defining properties, and magicians are only allowed to interact if they reside at the same or adjacent nodes. This does not require any reformulation of the fundamental equations given above, but can be incorporated in the filtering operator.

This kind of system may at first sound like an absolute, formless chaos. But this glib perspective ignores something essential – the phenomenon, well known for decades among European systems theorists (Varela 1978; Kampis 1991), of mutual intercreation or *autopoiesis*. Systems of magicians can interproduce. For instance, a can produce b, while b produces a. Or a and b can combine to produce c, while b and c combine to

produce a, and a and c combine to produce b. The number of possible systems of this sort is truly incomprehensible. But the point is that, if a system of magicians is mutually interproducing in this way, then it is likely to *survive* the continual flux of magician interaction dynamics. Even though each magician will quickly perish, it will just as quickly be re-created by its co- conspirators. Autopoiesis creates self-perpetuating order amidst flux.

Some autopoietic systems of magicians might be unstable; they might fall apart as soon as some external magicians start to interfere with them. But others will be robust; they will survive in spite of external perturbations. In (Goertzel 1995b) these robust magician systems are called **autopoietic attractors**. This leads up to the natural hypothesis that thoughts, feelings and beliefs are autopoietic attractors. They are stable systems of interproducing pattern/processes.

But autopoietic attraction is not the end of the story. The next step is the intriguing possibility that, in psychological systems, there may be a global order to these autopoietic attractors. In (Goertzel, 1994) it is argued that these structures must spontaneously self-organize into larger *autopoietic superstructures* – and, in particular, into a special attracting structure called the *dual network*.

The dual network, as its name suggests, is a network of magicians that is simultaneously structured in two ways. The first kind of structure is *hierarchical*. Simple structures build up to form more complex structures, which build up to form yet more complex structures, and so forth; and the more complex structures explicitly or implicitly govern the formation of their component structures. The second kind of structure is *heterarchical*: different structures connect to those other structures which are *related* to them by a sufficient number of pattern/processes. Psychologically speaking, as is elaborated in (Goertzel, 1993b; 1994), the hierarchical network may be identified with command-structured perception/control, and the heterarchical network may be identified with associatively structured memory. Mathematically, the formal definition of the dual network is somewhat involved; one approach is given in (Goertzel, 1995b). A simplistic dual network, useful for guiding thought though psychologically unrealistic, is a magician population living on a graph each node of which is connected to certain "heterarchical" neighbor nodes and certain "hierarchical" child nodes.

A *psynet*, then, is a magician system which has evolved into a dual network attractor. The core claim of the "psynet model" is that intelligent systems are psynets. This does not imply that all psynets are highly intelligent systems; one can build a simplistic implementation of the psynet model that runs on an ordinary PC, and certainly does not deserve the label "intelligent." What makes the difference between intelligent and unintelligent psynets is above all, I have argued, *size*. Small psynets do not have the memory or processing power required to generate self- and reality-theories. Thus they can never possess general intelligence.

Obviously size is not the whole story: the power and flexibility of the component magicians also plays a role in determining system intelligence. But a substantial number of magicians is certainly necessary, in order to support the hierarchical and heterarchical build-up of processes for "getting oriented," as described in the previous section. Self- and reality- theories, in the psynet model, arise as autopoietic attractors *within the context of the dual network*. They cannot become sophisticated until the dual network itself has self-organized to an acceptable degree. On the other hand, the dual network cannot grow to encompass extremely complex situations without the help of self- and reality-theories. There is a delicate symbiosis here which has never been seen to emerge from an AI program.

Until we understand the workings of the human brain, or build massively MIMD parallel "brain machines," the psynet model will remain in large part an unproven hypothesis. However, the intricate mathematical constructions of the logic-oriented AI theorists are also speculations. The idea underlying the psynet model is to make mathematical speculations which are psychologically plausible. Complex systems science, as it turns out, is a useful tool in this regard. Accepting the essential role of the self means accepting the importance of self-organization and complexity for the achievement of flexible, creative intelligence.

6 A-IS

The recognition of the cognitive importance of the self leads to a number of suggestions regarding the future direction of AI research. One of the most interesting such suggestions is the concept of $A-IS$, or "artificial intersubjectivity." The basis of A-IS is the proposition that self- and reality-theories can only evolve in an appropriate *social* context. While almost self-evident from the point of view of personality psychology, this proposition has been almost completely ignored by AI theorists. Today, however, computer science has progressed to the point where we can begin to understand what it might mean to provide artificial intelligences with a meaningful social context.

In principle, any artificial life world populated with intelligent agents could become an A-IS system, under appropriate conditions. The agents could come to collude in the modification of their world, so as to produce a mutually more useful simulated reality. In this way they would evolve interrelated self- and reality- theories, ergo artificial intersubjectivity. But speaking practically, this sort of "automatic intersubjectivity" cannot be counted on. Unless the different AI agents are in some sense "wired for cooperativity," they may well never see the value of collaborative subjective-world-creation. We humans became intelligent in the context of collaborative world-creation, of intersubjectivity (even apes are intensely intersubjective). Unless one is dealing with AI agents that evolved their intelligence in a social context – a theoretically possible but pragmatically tricky solution – there is no reason to expect significant intersubjectivity to spontaneously emerge through interaction.

Fortunately, it seems that there may be an alternative. I will describe a design strategy called "explicit socialization" or e.s., which involves explicitly *programming* each AI agent, from the start, with:

- an a priori knowledge of the existence and autonomy of the other programs in its environment
- an a priori inclination to model the behavior of these other programs.

In other words, in this strategy, one *enforces* A-IS from the outside, rather than, as in natural "implicit socialization," letting it evolve by itself. An initial implementations of e.s. is currently in the design stage.

To make the idea of explicit socialization a clearer, one must introduce some formal notation. Suppose one has a simulated environment $E(t)$, and a collection of autonomous agents $A_1(t)$, $A_2(t),\ldots,$ $A_N(t)$, each of which takes on a different state at each discrete time t. And, for sake of simplicity, assume that each agent A_i seeks to achieve a certain particular goal, which is represented as the maximization of the real-valued function $f_i(E)$, over the space of possible environments E. This latter assumption is psychologically debatable, but here it is mainly a matter of convenience; e.g. the substitution of a shifting collection of interrelated goals would not affect the discussion much.

Each agent, at each time, modifies E by executing a certain *action* $Ac_i(t)$. It chooses the action which it suspects will cause $f_i\left(E(t+1)\right)$ to be as large as possible. But each agent has only a limited power to modify E, and all the agents are acting on E in parallel; thus each agent, whenever it makes a prediction, must always take the others into account. A-IS occurs when the population of agents self-organizes itself into a condition where $E(t)$ is reasonably beneficial for all the agents, or at least most of them. This does not necessarily mean that E reaches some "ideal" constant value, but merely that the vector $(A_1,\ldots,A_N,E)$ enters an *attractor* in state space, which is characterized by a large value of the society wide average satisfaction $(f_1+\ldots+f_N)/N$.

The strategy of explicit socialization has two parts: *input* and *modeling*. Let us first consider input. For A_i to construct a model of its society, it must recognize patterns among the Ac_j and E; but before it can recognize these patterns, it must solve the more basic task of distinguishing the Ac_j themselves. In principle, the Ac_i can be determined, at least approximately, from E; a straightforward AILife approach would provide each agent with E alone as input. Explicit socialization, on the other hand, dictates that one should supply the Ac_i as input directly, in this way saving the agents' limited resources for other tasks. More formally, the input to A_i at time t is given by the vector $\left(Ac_{v(i,1,t)}(t),\ldots,Ac_{v(i,n(t),t)}(t),E(t)\right)$ for some

$n < N$, where the range of the index function $v(i,,)$ defines the "neighbors" of agent A_i, those agents with whom A_i immediately interacts at time t. In the simplest case, the range of i is always $1,\ldots,N$, and $v(i,j,t) = j$, but if one wishes to simulate agents moving through a spatially extended environment, then this is illogical, and a variable-range v is required.

Next, coinciding with this specialized input process, explicit socialization requires a contrived *internal modeling process* within each agent A_i. In straightforward AILife, A_i is merely an "intelligent agent," whatever that might mean. In explicit socialization, on the other hand, the internal processes of each agent are given a certain a priori structure. Each A_i, at each time, is assumed to contain $n(t) + 1$ different modules called "models":

- a model $M(E|A_i)$ of the environment, and
- a model $M(A_j|A_i)$ of each of its neighbors.

The model $M(X|A_i)$ is intended to predict the behavior of the entity X at the following time step, time $t + 1$.

At this point the concept of explicit socialization becomes a little more involved. The simplest possibility, which I call *first order e.s.*, is that the inner workings of the models $M(X|A_i)$ are not specified at all. They are just predictive subprograms, which may be implemented by any AI algorithm whatever.

The next most elementary case, *second order e.s.*, states that each model $M(A_j|A_i)$ *itself contains a number of internal models.* For instance, suppose for simplicity that $n(t) = n$ is the same for all i. Then second order *e.s.* would dictate that each model $M(A_j|A_i)$ contained $n + 1$ internal models: a model $M(E|A_j|A_i)$, predicting A_j's internal model of E, and n models $M(A_k|(A_j|A_i))$, predicting A_j's internal models of its neighbors A_k.

The definition of n'th order e.s. for $n > 2$ follows the same pattern: it dictates that each A_i models its neighbors A_j as if they used $n - 1$'th order e.s. Clearly there is a combinatorial explosion here; two or three orders is probably the most one would want to practically implement at this stage. But in theory, no matter how large n becomes, there are still no serious restrictions being placed on the nature of the intelligent agents A_i. Explicit socialization merely guarantees that the *results* of their intelligence will be organized in a manner amenable to socialization.

As a practical matter, the most natural first step toward implementing A-IS is to ignore higher-order e.s. and deal only with first-order modeling. But in the long run, this strategy is not viable: we humans routinely model one another on at least the third or fourth order, and artificial intelligences will also have to do so. The question then arises: how, in a context of evolving agents, does a "consensus order" of e.s. emerge? At what point does the multiplication of orders become superfluous? At what depth should the modeling process stop?

It would seem that the second order of modeling is probably out of reach for all animals besides humans and apes. In fact, if Uta Frith's (1989) psychology of autism is to be believed, then even autistic humans are not capable of sophisticated second-order social modeling, let alone third-order modeling. They can model what other people do, but have trouble thinking about other peoples' images of *them*, or about the network of social relationship that is defined by each person's images of other people. This train of thought suggests that, while one can simulate some kinds of social behavior without going beyond first order e.s., in order to get true social complexity a higher order of e.s. will be necessary. As a first estimate one might place the maximum order of human social interaction at or a little below the "magic number seven plus or minus two" which describes human short term memory capacity. We can form a concrete mental image of "Joe's opinion of Jane's opinion of Jack's opinion of Jill's opinion on the water bond issue," a fourth-order construct, so we can carry out fifth-order reasoning about Joe ... but just barely!

More speculations, perhaps too many speculations. But if intelligence requires self, and self requires intersubjectivity, then there may be no alternative but to embrace A-IS. Just because strong AI is possible does not mean that the straightforward approach of current AI research will ever be effective. Even with arbitrarily much processing power, one still needs to respect the delicate and spontaneous self-organization of psychological structures such as the self.

References

[1] Agha, (1988) *Actors*, Cambridge MA: MIT Press

[2] Barnsley, Michael (1988) *Fractals Everywhere*, New York: Addison-Wesley

[3] Deutsch, David (1985) Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer, *Proc. R. Soc. London A 400*, pp. 97- 117

[4] Devaney, Robert (1988) *Chaotic Dynamical Systems*, New York: Addison-Wesley

[5] Epstein, Seymour (1984) The Self-Concept: A Review and the Proposal of an Integrated Theory of Personality, in *Recent Advances in Pesonality Psycholoyg*, Englewood Cliffs: Prentice-Hall

[6] Erdmann, R (1988) *Boundaries in the Mind*, Hilldale NJ: Erlbaum

[7] Frith, Uta (1989) *Autism: Explaining the Enigma*, Oxford: Blackwell

[8] Goertzel, Ben (1993) *The Structure of Intelligence: A New Mathematical Model of Mind*, New York: Springer-Verlag.

[9] Goertzel, Ben (1993a) *The Evolving Mind*, New York: Gordon and Breach.

[10] Goertzel, Ben (1994), *Chaotic Logic: Language, Thought and Reality from the Perspective of Complex Systems Science*, New York: Plenum.

[11] Goertzel, Ben (1995), MAGIC WORLD: An Implementation of the Psynet Model of Mind/Brain, unpublished manuscript

[12] Goertzel, Ben (1995a), Chance and Consciousness, *Psychoscience*, Spring-Summer 1995

[13] Goertzel (1995b) *From Complexity to Creativity*, submitted to Plenum Press.

[14] Green, David and Terry Bossomaier (Editors) (1994), *Complex Systems: from Biology to Computation*, Amsterdam: IOS Press

[15] Hameroff, Stuart (1990), Ultimate Computing. Amsterdam: North-Holland.

[16] Kampis, George (1991) *Self-Modifying Systems in Biology and Cognitive Sciencem* New York: Plenum.

[17] Langton, Chris (Editor) (1992) *Artificial Life II*, New York: Addison-Wesley.

[18] Penrose, Roger (1987) *The Emperor's New Mind*, New York: Addison-Wesley

[19] Varela, Francisco (1978) *Principles of Biological Autonomy*, New York: Elsevier

Strong vs. Weak AI

Matjaž Gams
Jožef Stefan Institute, Jamova 39, 61000 Ljubljana, Slovenia
Phone: +386 61 17-73-644, Fax: +386 61 161-029
E-mail: matjaz.gams@ijs.si, WWW: http://www2.ijs.si/~mezi/matjaz.html

Keywords: strong and weak AI, principle of multiple knowledge, Church's thesis, Turing machines

Edited by: Xindong Wu
Received: May 15, 1995 **Revised:** December 13, 1995 **Accepted:** December 17, 1995

An overview of recent AI turning points is presented through the strong-weak AI opposition. The strong strong and weak weak AI are rejected as being too extreme. Strong AI is refuted by several arguments, such as empirical lack of intelligence in the fastest and most complex computers. Weak AI rejects the old formalistic approach based only on computational models and endorses ideas in several directions, from neuroscience to philosophy and physics. The proposed line distinguishing strong from weak AI is set by the principle of multiple knowledge, declaring that single-model systems can not achieve intelligence. Weak AI reevaluates and upgrades several foundations of AI and computer science in general: Church's thesis and Turing machines.

1 Introduction

The purpose of this paper is to present an overview of yet another turn-around going on in the artificial intelligence (AI) community, and to propose a border between the strong (old) and weak (new) AI through the principle of multiple knowledge.

To understand current trends in artificial intelligence, the history of AI can be of great help. In particular, it records ever recurring waves of overenthusiasm and overscepticism (Michalski, Tecuci 1993):

Early Enthusiasm or Tabula Rasa Craze (1955-1965)[1]

The first AI era was impressed by the fact that human brains are several orders of magnitude slower that computers (in transmission as well as coupling speed). Therefore, making a copy of a human brain on a computer would have to result in something ingeniously better. Three subjects were predominant: (1) learning without knowledge, (2) neural modeling (self-organizing systems and decision space techniques), and (3) evolutionary learning.

Dark Ages (1965-1975)

In the second epoch it became clear that the first approach yielded no fruitful results. There were strong indications that the proposed methods were unable to make further progress beyond solving a limited number of simple tasks. After funds for artificial intelligence research were deeply cut worldwide, new approaches were searched for. This era recognized that to acquire knowledge one needs knowledge, and initiated symbolic concept acquisition.

Renaissance (1975-1980)

Research in artificial intelligence continued despite cuts in funding, since it is a subject that will probably challenge human interest forever. Taking modest aims more appropriate to the level of current technology and knowledge sometimes produced even better results than expected. The characteristics are: (1) exploration of different strategies, (2) knowledge-intensive approaches, (3) successful applications, and (4) conferences and workshops worldwide.

[1] Years are rounded by 5. Note that there are different opinions regarding the exact periods.

AI Boom (1980-1990)

Artificial intelligence R&D produced a number of commercial booms such as expert systems. Literature, conferences, funds and related events have been growing exponentially for a few years. Superprojects like the CYC project and the Fifth Generation project were in full progress approaching final stages. Artificial intelligence was reaching maturity as indicated by: (1) experimental comparisons of AI methods and systems, (2) revival of non-symbolic methods such as neural networks and evolutionary computing, (3) technology-based fields gained attention – agents and memory-based reasoning, (4) computational learning theory, (5) integrated and multistrategy systems, and (6) emphasis on practical applications. However, no generally accepted intelligent (i.e. "truly" intelligent) system was in sight.

New AI Winter (1990-1995)

Major AI projects like the Fifth Generation project or the CYC project have not resulted in intelligent or commercially successful products. Overexpectations backfired again and criticism emerged, with two basic claims:
(1) There are several indications that intelligence can not be easily achieved on digital computers with existing approaches and methodologies[2].
(2) Today's computers as well as existing approaches basically do not differ much from those of 30 years ago (apart from being faster and having better storing capacities) and, therefore, are very unlikely to approach not only human-level but also any level of intelligence established by biological intelligent systems.

Possible consequences are profound: for example, if computers can not think, then quests for true intelligence on computers are as unrealistic as searching for perpetuum mobile. Another possible implication is as follows: if computers can nevertheless think and if the brightest minds have not been able to achieve intelligence in over 30 years on the best computers available, then they must have been trying in the wrong directions.

Funds for science in general, and AI in particular are decreasing as a long-term trend.

Invisible AI plus First Dawn Approaching? (1995-...)

Invisible AI produces working systems, although it has disappeared from the first pages of scientific journals. Software engineers are adding model-based diagnoses, rule-based modules and intelligent-interface agents on top of their conventional systems. AI techniques are invisibly interwoven with existing systems. It is not top AI science, but it works.

At the same time, bold new ideas are emerging, challenging the fundamentals of computer science as well as science in general – the Turing machine paradigm, Gödel's theorem and Church's thesis.

Pollock (1989) writes: "It represents the dream of AI since its infancy, but it is a dream that has faded in much of the AI community. This is because researchers in AI have made less progress than anticipated in achieving the dream."

In the words of Minsky (1991): "the future work of mind design will not be much like what we do today".

After this short overview of AI history, the AI mega projects FGCS and CYC are analysed in Section 2. The strong vs. weak AI issue is presented in Section 3, showing the basic differences between the two approaches and describing polarisations between their proponents. The line between strong and weak AI is proposed along the principle of multiple knowledge in Section 4. The principle presents a necessary condition for better performance and true intelligence in real-life domains. Fundamentals of AI and computer science are reexamined through the weak-AI viewpoint in Section 5, including the Turing test, Church's thesis, Gödel's theorem, and Turing machines.

2 AI Mega-Projects

2.1 The Fifth Generation Computer Systems (FGCS) Project

The FGCS project (Furukawa 1993; FGCS 1993) was the first research project in Japan to embrace international collaboration and exchange (around

[2]This viewpoint is close to the one presented by Penrose (1990) – we humans would recognise any true intelligence although different from the one we possess. Of course, there would be opinions that only humans possess intelligence even in the case when an intelligent computer passed all tests. However, at present there is no such system in sight and this is only an imaginary situation.

100 scientists involved). It created a frenzy in the developed countries, fearing that Japan is going to take the lead in another central technological area – new generation computers. As a result, several other projects were started, based on logic programming (LP), the core of the FGCS project. The project was heavily based on logic programming to bridge the gap between applications and machines. Several (some concurrent) versions of Prolog (e.g. KL1) were designed to support different levels, from the user-interface to machine language. The profound effect of LP is obvious even today, as it remains one of the central areas of computer research despite recent criticism[3].

The most crucial question posed is: is logic appropriate for real-life tasks? Obviously, it has several advantages, among them a very strict formal basis, and great expressive power. However, while it may be suitable for computers and formalists, it may not be so for humans and intelligent systems in general. Arno Penzias says: "Logic is cumbersome – that's why humans rarely use it." The logical approach effectively assumes that AI is a subset of logic and that intelligence and life can be captured in a global and consistent logic form[4]. According to logicism (Birnbaum 1992)[5], knowledge representation is independent of its use – quite opposite to the new AI approach based on biological and cognitive sciences.

The progress in both logic programming and AI areas as well as in the pursuit of general-purpose parallel computers has been modest but certainly not null. Although the Fifth Generation has not been able to compete with commercial products, the rest of the world listened to it. Japan has already launched the Sixth Generation project, based on real-life domains, neural networks, optical connections, and heavy parallelism.

[3]Just recently there have been substantial cuts in LP funding in Europe.

[4]One should be careful to distinguish between different kinds of logic. Fuzzy logic, logic of informal systems, and many-valued logic seem to be quite different from the logicism analysed here. Inductive logic programming (Bratko, Muggleton 1995) is another area that should not be identified with "pure" logic approach.

[5]Note that logicism cannot be directly identified with Nilsson's work (1991).

2.2 The CYC Project

The CYC project was started by Dough Lenat in 1984 as a ten-year project (Stefik, Smoliar 1993; Lenat, Guha 1990; Lenat 1995). Substantial funding was provided by a consortium of American companies. It is based on two premises: that the time has come to encode large chunks of knowledge into a meta-system encoding common-sense knowledge, and that explicitly represented large-scale knowledge will enable a new generation of AI systems. This "knowledge is power" (the Renaissance-era slogan) approach claims that by using huge amounts of knowledge, performance and intelligence of new generation AI systems will increase substantially. The intention is to overcome one of the biggest obstacles of existing AI systems, their brittleness (dispersed isolated systems working only on carefully chosen narrow tasks).

The CYC project addresses the tremendous task of codifying a vast quantity of knowledge possessed by a typical human into a workable system. Lenat estimates (1995) that they have entered 10^6 general assertions into CYC's knowledge base, using a vocabulary with approximately 10^5 atomic terms. CYC is intended to be able to give on-line sensible answers to all sensible queries, not just those anticipated at the time of knowledge entry. Lenat and Guha estimate that this will require at least ten million appropriately organized items of information, including rules and facts that describe concepts as abstract as causality and mass, as well as specific histographic facts. CYC includes a wide range of reasoning facilities, including general deduction and analogical inference. Reasoning is done through argumentation, through comparison of pro and con arguments.

CYC is the first project of its magnitude, and therefore represents a pioneering work. Several questions and problems were posed for the first time. The whole project has strong emphasis on pragmatism – to make something workable. There are four important design characteristics: (1) the language is first-order predicate calculus with a series of second-order extensions (2) frames are the normal (general) representation for propositions, (3) nonmonotonic inferences are made only when explicitly sanctioned by the user, and (4) knowledge acquisition and inference involve different languages between which translation is

automatic.

All knowledge in CYC is encoded in the form of logical sentences, and not in diagrams, procedures, semantic nets, or neural networks. The mechanism for managing uncertainty is not as common as Bayesian networks or reason maintenance systems. One of the interesting aspects in the CYC project is the distinction between epistemological and heuristic levels of representation. A user communicates with CYC in a high level epistemological language. CYC translates queries and assertions in this language into a lower-level heuristic notation, which provides a variety of specialized inference mechanisms corresponding to special syntactic forms.

According to the authors, success will be achieved if the system works and is used by different institutions for further research and development of new (generation of) expert and knowledge-based systems[6].

There have been several strange events related to the project from the start. For example, in the overview book by Lenat and Guha (1990), there are 22 publications, of which 7 were written by the head of the project (Lenat). In (Lenat 1995) there are only 9 publications, and only 4 of them were not (co)authored by Lenat. In addition and as pointed out by one of the anonymous referees, CYC's runtime behaviour as well as the assessment of the program in (Lenat, Guha 1990) is far too brief to be convincing.

Reviewers of the project (Stefik, Smoliar 1993) generally claim that it has not succeeded to the point proclaimed by the authors (although the project is not fully completed and the final evaluation has not been published yet). Lenat even claimed that machines will start learning by themselves when the CYC computer system becomes operational around 1994 (Lenat 1989). In 1995, it is becoming clear that nothing like that is going to happen. According to critics like Dreyfus (MTCM 1992), the CYC system is as dull as any other program[7].

[6]Authors of the project have changed success criteria and basic aims a couple of times during the last ten years, obviously trying to please public interest and accommodate scientific remarks. One of these "commercial" moves was quite probably the astronomic price of the CYC system.

[7]The anonymous referees of the paper seem to share the opinion that the paper could be even more critical of the project.

On the other hand, important new understandings were arrived at, some positive and some negative, which could be very useful for new projects. In Lenat's words (1995): (CYC) " is not a bumb on a log. It saddens me how few software-related projects I can say that about these days."[8]

2.3 CYC and FGCS – AI Dinosaurs?

The two projects have addressed several fundamental questions and come with modest and in some areas even with reasonable success. CYC has managed to encode a huge amount of knowledge and the Fifth Generation project resulted in tens of working computer systems (software plus hardware). Implemented systems have worked better than commercial ones on specific tasks. Their apparent commercial failure lies in the fact that commercial computer products such as new PC's and workstations are not only more general and applicable than the products of these huge R&D projects, but also the pace of their progress was and still is faster.

Being a pioneer has its dangers, yet one has to do it if we are to get anywhere. After all, AI is constantly changing in search for true discoveries, and in a great majority of questionnaires it is predicted a great future.

But in the eyes of public, both CYC and the Fifth Generation project have not fulfilled their promises. The relative failure revived the old hypotheses that classical symbolic AI may not be able to achieve intelligence on digital computers. In the words of Dreyfus (MTCM 1992): (classical symbolic) "AI is finished".

The analogy with dinosaurs lies in the fact that CYC and FGCS represent dominant approaches and achievements of the time, but their evolutionary line is at best shaky. "Hairy", weak AI systems will probably supplement formal ones.

In the author's opinion, basic research directions in the two projects mentioned could not produce intelligent systems at all. Both projects have adopted the computationally strong-AI approach instead of at least combining it with others, e.g. cognitive weak-AI. Both projects relied on a one-sided approach, disregarding the "new school of

[8]In my personal opinion, CYC has shown that common-sense knowledge is essential for any intelligent program. That brittle systems still dominating AI are not related to any true intelligence.

AI". This new approach claims that to design an intelligent system, one has to give it all properties of intelligent creatures: unity (i.e. multiple knowledge and multistrategy approach), intentionality, consciousness and autonomy along with generality and adaptability. However, doing this will be much more difficult than previously expected.

3 Strong and Weak AI

3.1 Description

The terms "weak" and "strong" AI were originally defined by Searle (1982); here, we shall introduce similar ones based on our viewpoints.

There are several terms attached to the old and still dominant AI: symbolic, classical, formalistic, and strong. The latest alludes to several versions of the strong AI thesis. More or less they all claim that it is possible to obtain intelligence by pure algorithmic processes regardless of technology or architecture.

By weak AI we denote:

- the negation of the strong AI thesis
- adopting knowledge from interdisciplinary sciences to upgrade the computational approach.

The extreme version of strong AI is termed strong strong AI, and the extreme version of weak AI weak weak AI. Whereas strong strong AI claims that even thermostats have feelings, weak weak AI claims that only humans can have feelings because they are the only beings with souls. Both extremes fall out of the scope of this paper.

There are several analyses of the strong-weak relations. Here, we present Sloman's gradations of the strong–weak scale (Sloman 1992). His vision of weak AI is based on architectural upgrades of Turing machines. In that sense he tries to avoid mentalism and cognitive sciences completely. Instead, he tries to upgrade the formalistic Turing-machines approach with engineering knowledge.

Sloman denotes the strongest thesis of AI as T_1. Each version T_n declares something about an Undiscovered Algorithm of Intelligence (UAI). T_1 is the strongest version, claiming that every instantiation of UAI has mental abilities – all that matters are data and algorithms – no time, rich execution mechanisms, meaning. However, abstract and statical structures can not have mental abilities. An often quoted example is the book of Einstein's brain. Supposedly, this book is no different than all the information and algorithms stored in Einstein's head. Indeed, hardly anybody would claim that any book itself – be it of Einstein's brain, Turing machines or anything else – is capable of thinking or speaking. A book on its own without any execution mechanism can not perform any action at all.

A slightly modified version of T_1 is T_{1a}: every time-instantiation of UAI has mental abilities. This eliminates the book case, but has other obvious flaws. For example, if we throw a bunch of paper sheets into the air we certainly do not get anything intelligent even in the case that by chance a new interesting story emerges. The execution mechanism must be in some sort of stronger causal relation. What about Searle's Chinese room? According to Sloman the causal relation between a book (formal syntactic structures) and Searle (the execution mechanism) is too weak. There can be no understanding and intelligence in such a loose connection.

T_2 is a further modified version, requiring sufficient reliable links between program and process. This is not a strong, but a vague, mild version. Sloman analyses the properties of links between program and process from the engineering point of view. In his view, one algorithm executed on a single processor can not emulate intelligence. The process must consist of many interleaving and intensively communicating subprocesses. The architecture of the Turing machine with one algorithm and one processor (executioner) can not provide intelligence.

The difference between physical (T_4) and virtual (T_3) parallelism is similar to that between one- and many-processor architectures. One algorithm, however complicated, is not sufficient for intelligence. Parallelism has to be at the same time fine- and coarse-grained. Minsky, Moravec and Sloman have presented various parallel architectures.

Parallelism is discussed in greater detail: T_{p1} enables intelligence with a simulated continuous environment. T_{p2} needs a serial processor with time-sharing. T_{p3} states that intelligent properties can be obtained through an appropriate ne-

twork of computers.

What if any machine relying on digital technology is incapable of reproducing intelligence? T_5 declares that at least in some subsystems supercomputing power is necessary, e.g. chemistry or biology. According to Sloman, even such discovery could be very valuable for focusing further research in AI.

T_1: abstract and statical procedures can reproduce mind
T_{1a}: time instantiation of T_1 can have mental abilities
T_2: links between programs and mechanisms
T_3: virtual parallelism
T_4: physical parallelism
T_5: super-computing powers

Figure 1: Sloman's strong (top) – weak (bottom) AI scale.

In Figure 1 we can see Sloman's gradation of the strong-weak AI paradigms.

There are several other directions of weak AI indicating that the new discipline is intensively searching for new discoveries. The general approach seems promising, yet it is not clear in which particular direction the discovery of true intelligence lies. For the time being it seems that new AI is strongly related to interdisciplinary sciences, especially biological and cognitive sciences. In the words of Edelman (1992): "Cognitive science is an interdisciplinary effort drawing on psychology, computer science and artificial intelligence, aspects of neurobiology and linguistics, and philosophy."

3.2 Strong vs. Weak AI

The strong AI thesis has been attacked by Dreyfus (1979), Searle (1982), Winograd (1991) , and Penrose (Penrose 1989; 1990; 1994). According to Sloman (1992), some practitioners of AI believe in the strong strong thesis. But that is a reason for criticising them, not AI. In any field there are the "naive, ill-informed, over-enthusiastic", according to Sloman. In Sloman's opinion, the main reason for such thinking is lack of appropriate training in philosophy.

Fair to say, the author of this paper was not much different a couple of years ago. After all, all students in computer sciences get acquainted with Church's thesis and Turing machines. After a while technical details fade away, and we are left with a frame in our memory declaring that anything that can be computed is executable by the Turing machine. And that it has been shown that the proof that the Turing machine can not solve "normal" (computable) problems cannot itself be computable (operational).

Since weak AI opposes the core of not only predominant AI but also some interpretations of postulates of computer science in general, it is of no great surprise that it has been successfully suppressed until recent years. The ideas of Winograd, Dreyfus or Searle were more or less rejected in the natural and engineering sciences community. But the discussion is becoming less and less one-sided in recent years.

One of the turn-arounds was a discussion regarding the Oxford professor Roger Penrose. He is one of the most famous mathematical physicists, with several discoveries from physics (e.g. regarding black holes with Hawking) and mathematics (e.g. how to tile a plane non-periodically with only two shapes). He wrote his first book "The Emperors New Mind: Concerning Computers, Minds, and the Laws of Physics" (1989) because he was astonished by a TV debate with strong AI supporters. The title of the book alludes to the emperor's invisible dress – everybody admires it, yet there is nothing to be seen. According to Martin Gardner's foreword, Penrose is "the child sitting in the third row, a distance back from the leaders of AI, who dares to suggest that the emperors of strong AI have no clothes."

In 1994, Joseph R. Abrahamson describes Penrose as one of those "who in the name of any one of a number of gods want to destroy rationality and science. It is important to be particularly aware when one of our attempts, in however subtle a manor, to suggest this magic should supplant or even be used to embellish reason and logic."

Based on old literature citations in Penrose's book, the predominantly strong AI community harshly attacked Penrose because of his obvious lack of knowledge of current AI activities. Even more, Penrose's arguments remain debatable even inside the weak AI community.

Yet, the criticism of classical AI failed. In a reply to Abrahamson's critique, Cronin (1994)

writes: (the old) "AI community has become an arcane, closed-minded, and theoretically incestuous field of computer science." Such words certainly did not encourage friendliness between the so antagonized communities; however, they might contain at least a grain of truth especially regarding close-mindedness[9]. Angell (1993, p.15) writes: "Do those AI people really think they can capture meaning with a logico-mathematical analysis?"

In a reply to Cronin, Abrahamson (1994b) softens his criteria, posing the limit at rejecting nonscientific approaches. In this way he does not directly reject mild versions of weak AI.

There are several well-established researchers in weak AI representing the major human factor why this new wave of weak AI was not rejected as before:

- Francis Crick is probably one of the most well-deserved researchers for introducing consciousness as a legitimate subject of science. He shared a Nobel Prize for the discovery of DNA's structure in 1953. As a neuroscientist, he wants to study consciousness through the brain's internal structure.

- Another Nobel Prize winner in weak AI is Gerald M. Edelman. He shared the prize in 1972 for research on antibodies. He is the author of neural Darwinism, a theory promoting competition between groups of neurons as the basis of awareness and consciousness.

- Brian D. Josephson won his Nobel Prize in 1973 for a special quantum effect (Josephson's junction). He proposes a unified field theory encapsulating mystical and psychic experiences.

- Maurice W. Wilkes is one of computer-science pioneers and the first person ever earning money for AI-related events. In the 1992 paper in Communications of ACM he presents the opinion that classical AI is getting nowhere in the last years in the sense that all computer systems today are totally unintelligent, and that according to empirical observations intelligence may be out of reach of digital computers.

4 Principle of Multiple Knowledge

In this section a line delimiting strong from weak AI is proposed, using the principle of multiple knowledge[10] (Gams, Križman 1991). The principle is seen as an attempt to define an AI analogy of the Heisenberg physical principle which divides the world of atomic particles from the world of macro particles. Previous related work is presented, e.g. in (Sloman 1992, Minsky 1987, Minsky 1991, Penrose 1994). Our work is presented in (Gams, Karba, Drobnič 1993).

Knowledge about domain properties can be utilised as a single system (model) or as two or more subsystems, each representing a different viewpoint on the same problem. Usually, each (sub)model represents at least a part of the external world.

The **'general' thesis of multiple knowledge** states (Gams, Križman 1991): in order to obtain better performance in real-life domains, it is generally better to construct and combine several models representing different viewpoints on the same problem than one model alone, if only a reasonable combination can be designed.

'Reasonable' combination means e.g. a combination designed by a human expert. 'Performance' means e.g. percentage of successfully solved tasks.

The **'strong' thesis of multiple knowledge** states that multiple semantic models are an integral and necessary part of intelligence in any machine or being.

In real-life domains a single model can not achieve as good performance as multiple models because each model tries to fit data and noise according to its own structure and therefore tries to impose its own view. During the construction phase, it is difficult to estimate which of the models has imposed the most appropriate structure for the unseen data, and different subparts of the measurement space are typically more suitable for different models. When combining or integrating

[9] It should be noted that AI and closely related fields are becoming more and more open to discussions. For example, see (Clancey 1993; Minsky 1991; Vera, Simon 1993).

[10] While the majority of sections in this paper represent an overview of the strong-weak AI relations, this section describes the author's personal opinion and contribution.

single models it is usually not too difficult to eliminate unsuccessful parts of models.

The general thesis of multiple knowledge implies that by constructing only one model it is practically impossible to achieve the same performance as by multiple models. In other words, although multiple models can be at any time (with more or less effort) transformed into one single model with the same performance as a set of models, in general it is not possible to construct such a single model in the process of learning without designing multiple models.

Integration of models after they are designed seems not only feasible but also sensible because of reduction in storage and classification time. In our experiments (Gams, Karba, Drobnič 1993), after integration a decrease in complexity and an increase in classification accuracy was observed.

4.1 Confirmations of the Theses

Attempts to confirm the theses of multiple knowledge were performed by:

- analogy with humans, e.g. expert groups performing better than single experts; analogy to the human brain, neural Darwinism; analogy with the architecture of human brain, especially regarding split-brains. A hypothesis is presented that the human race owes its success to the rise of multiplicity in their brains (Gazzaniga 1989; Crick 1994; Brazdil *et al.* 1991; Edelman 1991).
- Empirical learning, e.g. by analyses of PAC learning, which show that a combined system works better or the same as the best single system (Littlestone, Warmuth 1991); by practical measurements.
- Simulated models, indicating that in real-life domains significant improvements can be expected when combining a couple of the best systems (Gams, Bohanec, Cestnik 1994).
- Average-case formal models, indicating that in real-world domains combining has to be only a little bit better than by chance (success rate around 0.6) in order to produce improvements (Gams, Karba, Drobnič 1991).
- Related cognitive sciences, confirming similar ideas as the Principle although not presented in a technical form (Dennett 1991).
- Quantum physics, where the multiple-worlds theory (Dewitt 1973) enables computing in multiple universes (Deutch 1985; 1992) thus representing a possible theoretical background for the Principle.

One-model systems work, but are not as useful as many-model systems in real-life domains. If top performance matters, combining or integrating several systems generally seems to be advantageous regardless of additional costs in programming and computer time.

The strong version of the Principle represents one of the necessary conditions for true AI. It is neither sufficient nor the only necessary condition. However, it does substantially narrow the search space from single-model to many-model systems. For example, over 99% of all existing computer systems and most current AI orientations are based on a single model. Intelligent systems seem to have special properties, e.g. multiplicity. These systems are very rare among all the systems. It is highly unlikely that we find (construct) them when searching in the space of all possible systems without correctly assuming their special properties.

The Principle is sometimes getting accepted as "everybody-knew-it-all-the-time". Indeed, there are many similar ideas around, e.g. Minsky's multiple representations (1991) or Sloman's parallel architectures (1992). Angell (1993, p. 15) writes: "As if every word were not a pocket into which now this, now that, now several things at once have been put!" Accepting the Principle means introducing weak AI and leads to fundamental changes in future progress in AI and computer science alike.[11]

5 Fundamentals of AI and Computer Science

Weak AI reexamines and disputes the soundness of several well-established scientific fundamentals: Turing's test, Gödel's theorem, Church's thesis, and the Turing machine.

[11] According to the Principle, many research directions will not produce true intelligence, meaning that efforts, achievements and future funding in that areas are doubtful.

5.1 Turing's Test

When Turing nearly half a century ago posed his famous question "Can computers think", electronic computers were just emerging. The back-bone of his test is a detective probabilistic quiz in which an interrogator has to be sufficiently sure which of the two subjects communicating through a computer interface (terminal plus keyboard) is human and which computer, given limited time. Turing believed that his test would be passed in around 50 years when computer storage capacity reached 10^9. By then, "an average interrogator would not have more than 70 per cent chance of making the right identification (as between human and computer) after five minutes of questioning."

During years, several modifications of Turing's test have being proposed, e.g. the total Turing test (TTT) in which the subject has to perform tasks in the physical world such as moving blocks. Other remarks imply that the original test is (1) too easy since it is based on typed communication only, (2) too narrow since it is basically an imitation game, (3) too brittle since it can not reveal the internal structure of thinking processes – Searle's basic claim (Searle 1982), and (4) too difficult since no animal and many humans (e.g. handicapped) are unable to compete at all, and intelligence can be displayed well below average-human level. All these remarks have their counterarguments, e.g. that (1) communication through typing is more than relevant to evaluate the intelligence of a subject, e.g. by the IQ tests, (2) such communication allows very rich possibilities of questions and themata, (3) it is not possible to reveal the human thinking process either, and (4) if the Turing test (TT) is too difficult then the limited Turing test (LTT) can be applied. Indeed, such is the case in practical contests held annually (Shieber 1994). TT remains probabilistic, approximate, detective, fundamentalistic, behaviouristic and functional.

Although the Turing test is heavily analysed and disputed, it remains the most interesting scientific test up to date, offering important implications.

The latest analyses of the Turing test were performed by Turing's contemporary Donald Michie (1993). In his opinion, there are two obstacles an intelligent computer system has to face in order to approach passing it:

1. subarticulacy – the human inability to articulate specific activities although performed by humans, and

2. superarticulacy – the ability to explain particular thought processes in a suitably programmed machine although being subarticulate in humans.

Regarding the first point, humans can not articulate their internal thought processes, which are sometimes more transparent to observers than to themselves. Therefore, how can human knowledge be transformed into computer systems if humans are not able to specify it?

The second point poses another problem. Computer programs are by default traceable – meaning their decisions can be traced and reproduced. Even systems like neural nets or numerical procedures can be 'understood' up to a point, and simulated by other transparent systems. All computer systems, therefore, have abilities nonexistent in humans.

Some of these questions were discussed already by Turing. He proposed that machines would have to play the imitation game, thus simulating thought processes while inherently being different. While it is not yet clear whether digital machines can achieve intelligence at all, it is becoming accepted that on digital computers, systems simulating human thought processes will be essentially different from humans. In light of this conclusion, the claim of connectionists – that sufficiently complex neural networks will be effectively the same as the human brain – is hard to accept. Even if neural networks were to achieve the performance of a human brain, it would be possible to extract weights, topology and other characteristics of nets. By not being able to do it in humans, one (of many) unavoidable substantial difference appears. The "End of Innocence" period, together with empirical verification, brings new insights, displaying the naivete of existing approaches and opening new directions. The Turing test indicates substantial differences between formal machines and real-life beings.

Weak AI is in general satisfied with less than passing the Turing test. For example, artificial life and evolutionary computing try to simulate rather primitive forms of life. Brooks (1991) proposes intelligence without reasoning,

low-intelligence robots (insects) without symbolic internal representation of the external world. Sloman (1992) finds the Turing machine rather unrelated to real life. It represents an artificial machine very capable for specific formal tasks only. Sloman, Penrose and also people in general tend to believe that even animals can display certain aspects of intelligence when solving real-life problems. On the other hand, while machines can solve difficult formal problems which are often practically unsolvable even by humans and definitively unsolvable for all animals, they are still regarded as totally unintelligent.

5.2 Church's Thesis and Turing Machine

Around 1930 Church, Gödel, Kleene, Post, Turing and others tackled questions such as: what can be computed and what not, are all statements either provable or not inside a formal system? They have come with basic concepts that represent a backbone of today's computer science.

Church's thesis is the assertion that any process that is effective or algorithmic in nature defines a mathematical function. These functions form a well-defined class, denoted by terms such as recursive, λ-definable, Turing computable. All these functions are computable by the Turing machine, a formal model of computers. Anything that a digital or analog computer can compute, be it deterministic of probabilistic, is computable by the abstract Turing machine, given enough time and space. The problems that the Turing machine can not solve are unsolvable for present and future formal computer systems as well, be it simple PC's, supercomputers or parallel connectionist machines.

Church's thesis provides the essential foundation for strong AI. If computable problems are solvable by the Turing machine then digital computers can solve them if only they are quick enough. Therefore, achieving true intelligence on computers demands only very fast hardware with sufficient memory capabilities and a program. In Abrahamson's opinion (1994) it is only a matter of time and technological progress.

In general, there are two major philosophical orientations regarding the human mind and our world in general: mentalistic and mechanistic. Mechanicists regard mind as a material object obeying the laws of nature. Mind is a (biological, physical ...) machine. Mentalists see mental states as something beyond formal sciences (mild version) or even extramaterial, i.e. outside the real world (strong version). Church's thesis implies that its computational essence can not be refuted by effective means. It means that the opposing hypothesis can not be effective at all, or in other words, it can not be computed in the general meaning of the word.

The strong principle of multiple knowledge collides with the direct explanation of Church's thesis. One possible compromise is that although intelligent models can be – at least in principle, with unknown practical problems – designed and executed on any Turing machine, it is not possible to design intelligent computer programs in the form of a single model not consisting of multiple models. Therefore, if the program on the Turing machine is multiple enough and has the needed additional properties, it could simulate intelligence. However, the principle does not exclude the other possibility – that true intelligence can not be achieved on Turing machines at all, that stronger computational mechanisms having explicit multiplicity at the core of the computing process are necessary.

Practically all weak AI researchers in this or another way distance their ideas from Church's thesis (see Section 3). Neuroscientists (Edelman 1992) propose their models of the brain. Physicists propose new physical theories enabling new computing mechanisms – Penrose proposes microtubules where quantum effects in relation to the correct quantum gravity enable supercomputing powers. Deutch (1992) proposes a quantum Turing machine.

Sloman's viewpoint is similar to the principle of multiple knowledge based on the engineering architecture of the computing machine. Theoretically, it has been proven that the computational power of one Turing machine is equal to the power of many parallel machines. From the engineering point of view this is not the case. The key is not in speed or time, but in the architecture. For example, a fatal error in one processor simulating parallel computing causes malfunction in serial architectures yet is usually only a smaller obstacle in appropriate parallel hardware architectures. If one processor simulates several virtual processors

then it must constantly check the internal states of each parallel process. This disables true asynchronous interaction with complex real-life environments. Although the parallel and sequential process display equal computational powers, they substantially differ in causal relations.

5.3 Seeing the Truth of Gödel's Sentence

In his 1931 paper, Gödel showed that for any formal system F broad enough to express the arithmetic of natural numbers, there is a construction of a formula $P_k(k)$ where k is the Gödel's number of that formula itself. This well-defined formula is denoted by $G(F)$. Gödel's theorem states that if F is consistent, there can be no derivation of $G(F)$, and if F is omega-consistent, no derivation of $\neg G(F)$. Therefore, $G(F)$ is undecidable (unprovable), and the formal system F is incomplete.

Not only that Gödel's theorem is formally provable, computer programs such as SHUNYATA (Ammon 1993) have been able to automatically reproduce, i.e. rediscover the proof.

By proving his theorem Gödel demolished the strong formalistic approach in science. He proved that at least one formula (statement, sentence) can not be proven inside a formal system (later it was found that there are many such statements). Therefore, there is no way a formal machine can prove a specific sentence constructed by a formal (legal) procedure.

Many relevant researchers including Gödel and Turing thought that although the proof shows that it is not possible to formally prove $G(F)$, $G(F)$ is nevertheless true. Of course, no formal proof of $G(F)$ can be constructed inside F since it has been formally proven that such a proof does not exist. Therefore, how can $G(F)$ be seen as true by humans? In 1961 Lucas presented his view of this paradoxical situation hypothesising what happens if humans use some kind of a formal algorithm UAI. This idea was revived and extended by Penrose (1989).

Lucas proposes – in his viewpoint – a valid mathematical procedure for seeing the truth of $G(F)$. Namely, if the sentence asserts about itself that it is not provable, and the formal proof showed that $G(F)$ can not be proved, then the sentence is obviously true. Therefore, humans can see at $G(F)$ is true.

Penrose's extension is as follows: even if a human uses some kind of (probably very complex) formal algorithm UAI executable on a Turing machine, and we construct a formal Gödel's sentence $G(UAI)$ for that algorithm, he can see the truth of it. Not only Penrose and mathematicians, probably all students in natural and technical sciences can intuitively see (or have that feeling of) the truth of Penrose's line of reasoning. Therefore, we can assume that all humans are at least in principle able to see it. Furthermore, all humans use similar processes when seeing the truth of Gödel's sentence.

Since formal systems are not able to formally prove the truth of Gödel's sentence, and humans can see it, humans do not always apply formal algorithms (e.g. UAI). Therefore, since humans can in principle reproduce anything that Turing machines can, and Turing machines in principle can not reproduce all things humans can (e.g. seeing the truth of Gödel's sentence), Turing machines do not possess all computational powers that humans do. Since Turing machines are capable of reproducing any computation by digital computers, true intelligence can not be achieved on digital computers.

Among the common objections to this kind of reasoning are the following:

- it is not possible to see that $G(F)$ true since this requires proving that F is consistent[12];
- $G(F)$ can be seen to be true by flible and incomplete procedures (similar to the ones humans use);
- Gödel's theorem is not related to real life; it is just a formal matter relevant to formal systems. Although this means that we have to reject deductive semantics as means of describing human intelligence, we can endorse other types of inference, e.g. abductive logic.
- in a computationally stronger meta$F$ it is possible to formally prove a statement (theorem) $provable(metaF, G(F))$.

[12] As pointed out by Boolos, Chalmers, Davis and Perlis (Penrose 1990), the consistency of complex mathematical systems, e.g. ZF systems, can not be proved. This means that nobody, Turing machines and Penrose included, can prove or even see the truth of Gödel's sentence in ZF systems.

The most fundamental denial of Penrose's argument was presented by Sloman (1992). He attacked the core meaning of Gödel's theorem: Gödel's sentence does not mean what it seems to mean, and Penrose can not see the truth of $G(F)$ since there are models in which it is true and those in which it is false.

The first premise does not seem to be justified as shown by Bojadžiev (1995).

Sloman's claim is based on constructing two models: of $(F, G(F))$ and of $(F, \neg G(F))$. This is valid since neither $G(F)$ nor $\neg G(F)$ are provable in F, if consistent. Now, nobody can see the truth of $G(F)$ in $(F, \neg G(F))$, Penrose concluded.

However, in models of $(F, \neg G(F)$ it is possible to establish the truth of $\neg G(F)$, therefore, $G(F)$ is not unprovable anymore if $(F, \neg G(F)$ is consistent. Extended models of F usually do not correspond to classes of universal Turing machines. This is a common case in computational capabilities of systems: stronger mechanisms can often answer puzzles in weaker mechanisms, yet have their own undecidable questions. Sometimes it is even sufficient to apply meta–reasoning inside systems with the same computational powers, but again new undecidable questions can be produced. For example, it has been formally proven by a meta-system that Gödel's sentence $G(F)$ is true in natural numbers if F is consistent. Therefore, the truth of Gödel's theorem in certain mathematical models, e.g. in Peano Arithmetic can be formally proven outside F if it is consistent.

Here we shall translate the same problem into the world of Turing machines. Namely, Gödel's theorem corresponds to the halting problem of Turing machines, i.e. to the question if a Turing machine can in general predict whether a Turing machine will stop or not. It has been formally proven that the halting problem is in general undecidable (Turing 1936; Hopcroft, Ullman 1979). Furthermore, the concept of Gödel's theorem is so fundamental for formal systems that it can be reproduced in many forms (see for example Penrose's second book (1994)).

Consider for example an Algol-like procedure U which shows that a procedure can not determine whether it will stop or not. Reasoning starts with the hypothesis that there exists a procedure T which can determine for any procedure *proc* whether it stops or not. Then we construct a procedure U which includes the procedure T. If U itself (self-reference) is given as an input for U, it should stop when it should not (i.e. $T(U)$ is $false$) and vice versa. Since the transformation from T to U is legal inside the same description mechanism of Turing machines, and U cannot exist, T cannot exist. Therefore, a procedure which determines for any procedure whether it will stop or not, does not exist.

```
procedure U(proc);
begin
        while T(proc) do;
        write('OK');
end;
```

The self-referential applicability of U, and the halting problem in Turing machines and formal programming languages are beyond reasonable doubt. Furthermore, high-school students usually do not have troubles seeing or understanding the paradoxical nature of the halting problem.

Penrose replies that there is no reason for dealing with unsound or incomplete systems. Under this assumption it is possible to see the truth of Gödel's sentence, it is possible to formally prove it outside F, and quite probably possible to duplicate Penrose's semantical reasoning about truth by special meta-systems.

In summary, Penrose's version of the Gödel theorem and the halting problem represents an interesting hypothesis, however is not proven. On the other hand, several attempts to formally disprove Penrose's version have been formally proven to be wrong.

6 Discussion

The history of AI teaches us that the only constant is its ever-changing nature. In recent years new, fresh ideas are coming from interdisciplinary sciences – neurobiology, philosophy, cognitive sciences. In this way, the computational approaches are being enriched and upgraded.

Weak AI reexamines basic postulates of AI and computer science. In regard to *Turing's test*, proponents of weak AI see the test as an indicator of important differences between humans and computers. Computer systems can explain their line of reasoning in detail. Humans do not know how

reasoning is performed in their heads and do not know how to reveal (transplant) that to computers. Just passing the test is not sufficient to be accepted as intelligent. A computer chess program beating most humans is not intelligent although it performs brilliantly compared to an average human. Animals are not capable of playing chess, yet some of them show properties of intelligence while computers are regarded as totally unintelligent.

By-passing *Church's thesis*, weak AI does not accept that one *Turing machine* performing one algorithm is sufficient to achieve intelligence. The *principle of multiple knowledge* proposes multiple-model structures as one of necessary conditions for intelligent systems. Extreme viewpoints see digital computers as incapable of achieving intelligence.

There are several indications that the human brain is computationally more powerful than digital computers, e.g. observed through the progress of computer power and the lack of computer intelligence. Theoretical analyses are often performed through the *Gödel theorem* and *halting problem.*

The principle of multiple knowledge dictates a step-up of complexity from one optimal model to an optimal combination of models. It upgrades the centuries old *Occam's Razor* indicating that the Razor can be even misleading when blindly applied. However, an upgraded version of Occam's Razor might be valid in the multiple-model world. Similarly, human knowledge is seen as significantly more complex than currently expected. Multiple models introduce an additional level of combinatorial explosion, thus making knowledge less transparent, more difficult to store, and more powerful.

Clashes between strong and weak AI proponents may help sift new ideas and eliminate unsound attempts. Weak AI is still in the brainstorming state – lots of new ideas and not many confirmed achievements. Weak AI is getting accepted as another discipline researching consciousness and relations to computers. Similarly, most of new nonsymbolic approaches in AI were rejected at first and then accepted, be it neural networks or evolutionary computing.

How can weak AI be proven wrong? The simplest proof would be constructive – to design a single-model computer system capable of true intelligent behaviour. Note that just designing an intelligent computer system executable on a Turing machine is not enough.

How can strong AI be proven wrong? There are several possibilities. For example, it is enough that the Penrose's hypothesis about Gödel's theorem gets proven. Or that the principle of multiple knowledge gets proven. Or that neuroscience produces substantial new discoveries about the human brain. Or that a new physical theory gets proven. Or ...

Today, the house of science is based on empirical validation and formal verification. Formal verification is well within the domain of Turing computable functions. The fear that weak AI is attacking the core of science by reevaluating Church's thesis and other scientific postulates is not grounded. For example, if Penrose's ideas get accepted, meaning that unprovable true functions are computable by humans but not by computers, scientific knowledge will essentially expand. Science will expand even if the principle of multiple knowledge gets accepted. Instead of relying on formal models, other aspects will gain prominence, e.g. engineering or cognitive enrichments of formal sciences.

Acknowledgments

This work was supported by the Ministry of Science, Research and Technology, Republic of Slovenia and was carried out as part of different European projects. Research facilities were provided by the "Jozef Stefan" Institute. The author is indebted to the two anonymous reviewers and in particular the editor Xindong Wu for their constructive comments on the paper. Special thanks to Damjan Bojadžiev for careful reading and correcting the paper, and to Mare Bohanec, Matija Drobnič, Nada Lavrač and Donald Michie for helpful remarks.

References

[1] K. Ammon (1993), An Automatic Proof of Gödel's Incompleteness Theorem, *Artificial Intelligence*, **61**, pp. 291-307.

[2] I. O. Angell (1993), Intelligence: Logical or Biological, Viewpoint, *Communications of the ACM* **36**, pp. 15-16; 110.

[3] J. R. Abrahamson (1994), Mind, Evolution, and Computers, *AI magazine*, Spring 1994, pp. 19-22.

[4] J. R. Abrahamson (1994b), A Reply to Mind, Evolution, and Computers, *AI magazine*, Summer 1994, pp. 8-9.

[5] L. Birnbaum (1992), Rigor Mortis: A Response to Nilsson's "Logic and Artificial Intelligence", *Foundations of artificial intelligence*, pp. 57-79, (ed.) D. Kirsh, MIT/Elsevier.

[6] D. Bojadžiev (1995), Sloman's View of Gödel's Sentence, *Artificial Intelligence* **74**, pp. 389-393.

[7] I. Bratko and S. Muggleton (1995), Applications of Inductive Logic Programming, *Communications of the ACM* **38**, pp. 65-71.

[8] P. Brazdil, M. Gams, L. Sian, L. Torgo and W. van de Velde (1991), Learning in Distributed Systems and Multi-Agent Environments, *Proc. of EWSL-91*, Porto, Portugal.

[9] R. A. Brooks (1991), Intelligence without Representation, *Artificial Intelligence* **47**, pp. 139-160.

[10] W. J. Clancey (1993), Situated Action: A Neuropsychological Interpretation Response to Vera and Simon, *Cognitive Science* **17**, pp. 87-116.

[11] F. Crick (1994), *The Astonishing Hypothesis, The Scientific Search for the Soul*, New York.

[12] M. R. Cronin (1994), A Reply to Mind, Evolution, and Computers, *AI magazine*, Summer 1994, p. 6.

[13] D. C. Dennett (1991), *Consciousness Explained*, Little Brown.

[14] D. Deutch (1985), Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer, *Proceedings of Royal Society*, pp. 97-117.

[15] D. Deutch (1992), Quantum Computation, *Physics Word*, pp. 57-61.

[16] B. S. Dewitt (1973), *The Many-Worlds Interpretation of Quantum Mechanics*, Princenton University Press.

[17] H. L. Dreyfus (1979), *What Computers Can't Do*, Harper and Row.

[18] G. Edelman (1992), *Bright Air, Brilliant Fire, On the Matter of the Mind*, Penguin Books.

[19] FGCS (1993), The Fifth Generation Project, *Communications of the ACM* **36**, pp. 46-100.

[20] K. Furukawa (1993), Fifth Generation Computer Systems (FGCS) Project in Japan, *Japan Computer Quarterly* **93**, pp. 1-33.

[21] M. Gams, M. Bohanec and B. Cestnik (1994), A Schema for Using Multiple Knowledge, *Computational Learning Theory and Natural Learning Systems*, Vol. 2, MIT Press, pp. 157-171.

[22] M. Gams, M. Drobnič and M. Petkovšek (1991), Learning from Examples – A Uniform View. *Int. Journal for Man-machine Studies* **34**, pp. 49-89.

[23] M. Gams, N. Karba and M. Drobnič (1993), Average-Case Improvements when Integrating ML and KA, *Proc. of IJCAI'93 Workshop: Machine Learning and Knowledge Acquisition*, France, pp. 79-95.

[24] M. Gams and V. Križman (1991), The Principle of Multiple Knowledge, *Informatica* **15**, pp. 23-28.

[25] M. S. Gazzaniga (1989), Organization of the Human Brain, *Science* **245**, pp. 947-952.

[26] J. E. Hopcroft and J. D. Ullman (1979), *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley.

[27] D. B. Lenat (1989), When Will Machines Learn?, *Machine Intelligence* **4**, pp. 255-257.

[28] D. B. Lenat (1995), CYC: A Large-Scale Investment in Konwledge Infrastructure, *Communications of the ACM* **38**, pp. 33-38.

[29] D. B. Lenat and R. V. Guha (1990), *Building Large knowledge-Based Systems: Representations and Inference in the Cyc Project*, Addison-Wesley.

[30] N. Littlestone and M.K. Warmuth (1991), The Weighted Majority Algorithm, Technical Report UCSC-CRL-91-28, USA.

[31] J. R. Lucas (1961), Minds, Machines and Gödel, *Philosophy* **36**, pp. 112-127.

[32] R. S. Michalski and G. Tecuci (1993), Multistrategy Learning, *IJCAI-93 Tutorial T15*, Chambery, France.

[33] D. Michie (1993), Turing's Test and Conscious Thought. *Artificial Intelligence* **60**, pp. 1-22.

[34] M. Minsky (1987), *The Society of Mind*, New York: Simon and Schuster.

[35] M. Minsky (1991), Society of Mind: A Response to Four Reviews, *Artificial Intelligence* **48**, pp. 371-396.

[36] MTCM (1992), The Machine that Changed the Word, TV series, ACM.

[37] N. J. Nilsson (1991), Logic and Artificial Intelligence, *Artificial Intelligence* **47**, pp. 31-56.

[38] R. Penrose (1989), *The Emperor's New Mind: Concerning computers, minds, and the laws of physics*, Oxford University Press.

[39] R. Penrose (1990), Precis of the Emperor's New Mind: Concerning computers, minds, and the laws of physics, *Behavioral and Brain Sciences*, **13**, pp. 643-705.

[40] R. Penrose (1994), *Shadows of the Mind, A Search for the Missing Science of Consciousness*, Oxford University Press.

[41] J. L. Pollock (1989), *How to Build a Person: A Prolegomenon*, MIT Press.

[42] J. R. Searle (1982), The Chinese Room Revisited, *Behavioral and Brain Sciences* **8**, pp. 345-348.

[43] S. M. Shieber (1994), Lessons From a Restricted Turing Test, *Communications of the ACM* **37**, pp. 70-78.

[44] A. Sloman (1992), The Emperor's Real Mind: Review of the Roger Penrose's The Emperor's New Mind: Concerning Computers, Minds and the Laws of Physics, *Artificial Intelligence* **56**, pp. 335-396.

[45] M. J. Stefik and S. W. Smoliar (ed.) (1993), The Commonsense Reviews, *Artificial Intelligence* **61**, pp. 37-181.

[46] A. M. Turing (1936), On Computable Numbers with an Application to the Entscheidungsproblem, *Proc. London Math. Soc.* **2**, pp. 230-265.

[47] A. H. Vera and H. A. Simon (1993), Situated Action: A Symbolic Interpretation, *Cognitive Science* **17**, pp. 7-48.

[48] M. W. Wilkes (1992), Artificial Intelligence as the Year 2000 Approaches, *Communications of the ACM* **35**, pp. 17-20.

[49] T. Winograd (1991), Thinking Machines: Can there be? Are We?, *The Boundaries of Humanity: Humans, Animals, Machines*, Berkeley, University of California press, pp. 198-223, (ed.) J. Sheehan, M. Sosna. Also in this issue.

A Brief Naive Psychology Manifesto

Stuart Watt
Department of Psychology, The Open University,
Milton Keynes MK7 6AA. UK.
Phone: +44 1908 654513; Fax: +44 1908 653169
E-mail: S.N.K.Watt@open.ac.uk

Keywords: naive psychology, common sense, anthropomorphism

Edited by: Matjaž Gams
Received: May 11, 1995 **Revised:** October 2, 1995 **Accepted:** October 23, 1995

This paper argues that artificial intelligence has failed to address the whole problem of common sense, and that this is the cause of a recent stagnation in the field. The big gap is in common sense—or naive—psychology, our natural human ability to see one another as minds rather than as bodies. This is especially important to artificial intelligence which must eventually enable us humans to see computers not as grey boxes, but as minds. The paper proposes that artificial intelligence study exactly this—what is going on in people's heads that makes them see others as having minds.

1 Introduction: from naive physics to naive psychology

Ten years ago, Hayes published the "Second naive physics manifesto" (Hayes, 1985b). Hayes proposed that we "put away childish things by building large-scale formalisations," beginning with "our knowledge of the everyday physical world." He, and others, have since put a lot of effort into developing models of our common sense understanding of the physical world.

But common sense has been a big problem for artificial intelligence, and despite the attempts of many brave souls (e.g. Hayes, McCarthy, McDermott, and Lenat) it hasn't really yielded: "the common sense knowledge problem has blocked all progress in theoretical artificial intelligence" (Dreyfus & Dreyfus, 1988). This is due to deep technical and methodological problems which have arisen in the study of common sense, most famous of which is perhaps the "frame problem" (McCarthy & Hayes, 1969).

This paper will present a position orthogonal to that of Hayes. Hayes' initiative clarified many of the issues associated with common sense, and other developments in comparative and developmental psychology have further highlighted the apparently fundamental nature of naive physics but they have also revealed a deeper and bigger problem than that of naive physics—naive psychology.

Naive psychology (Clark, 1987, Hayes, 1985b) can be thought of as the natural human ability to infer and reason about other people's mental states—the faculty that normal adult people have which, in short, enables them to see one another as minds rather than bodies. This is an issue that artificial intelligence must also address. Although people see one another as minds not simply bodies, they don't see computers as minds in the same way (Caporael, 1986). To overcome this barrier, we humans must be able to see minds in artifacts, to ascribe mental states to artificial intelligences in the same way that we do to people.

There is evidence that a lot of human intelligence is 'Machiavellian' (Byrne & Whiten, 1988) in the way people use it to outwit each other and to recognise and manipulate one another's mental states; our social environments are considerably more complex than our physical ones. Survival in these social environments require us to become "natural psychologists" (Humphrey, 1984), capable of recognising and reasoning about one another's mental states. Naive psychology is at the core of our understanding of the world. Humphrey even suggests that naive physics may be

itself derived from a leaky naive psychology.

At the heart of this proposal is a methodological inversion. Usually, artificial intelligence is thought of as the 'science of smart behaviour'—building systems which behave in a way that seems 'intelligent.' This leads to all sorts of navel-oriented definitions and operational interpretations of the word 'intelligent,' none of which help to find intelligence. The reason they don't help is they miss the point: a definition of intelligence becomes part of science, but doesn't have any impact where it counts, which is on everyday human naive psychology. Artificial intelligence also needs to study naive psychology to find out what is going on in my head to make me see other people as having minds—and it is this that is an inversion of the conventional approach. There should be two parts to the study of intelligence: the smart behaviour we're all familiar with, but also our ability to *recognise* that behaviour as smart.

So the sin of artificial intelligence is a sin of omission—it hasn't properly addressed the second part of the problem, that of naive psychology. Naive psychology is not more important or significant than other abilities, but it is equally an essential element of human cognition, and, further, it is an important part of how we recognise intelligence. It should become a topic for serious research in artificial intelligence.

Naive psychology isn't new to artificial intelligence, which has already tried a number of approaches to the problem. Perhaps the most successful have been the axiomatic formalisms (e.g. Cohen & Levesque, 1990), which represent naive psychology as the ability to make inferences about a set of beliefs, desires, and intentions, corresponding to an agent's mental states. These axiomatic formal approaches to naive psychology are usually based on some kind of modal logic. These logics enable representation and reasoning about someone's mental states by making these states unobservable, so agents can believe something which other people know to be false, for example.

But representational approaches to naive psychology also have their critics (e.g. McDermott, 1987). McDermott's criticism is that the representations and the use of those representations cannot be truly separated, as they are separated in formalisms based on pure logic. But besides this criticism, there are deeper ones—are mental states really best described in terms of beliefs, desires, and intentions? (Dennett, 1987). This is an assumption, and while it works well a lot of the time, there are some mental states which fit uneasily in this model (e.g. moods, hostility.)

The most widely voiced criticisms of explicit representations in artificial intelligence haven't really had much impact in this field, because they have to deal with something rare in physical environments, this opacity that people have. Both situated and connectionist approaches break down with the opaque nature of other agents. While these approaches are often good at dealing with observables, they are less good at dealing with the unobservable nature of other agent's mental states. This really does seem to require something which does the job of a theory—and this is what representations are good at. If representations had to become situated to deal with the physical world, situated approaches have to become representational to deal with the psychological one.

There has been significant work in this field, but perhaps artificial intelligence just hasn't realised the scale or importance of the problem. All the major philosophical stumbling blocks of artificial intelligence (e.g. consciousness, intentionality) can be traced to our inability to understand when to ascribe mental states to computers or other artifacts. This doesn't mean that naive psychology is logically prior to these problems, but that it is methodologically prior.

Within artificial intelligence, there is often an assumption that there is something which can be called 'intelligence,' but which is very different from what we call 'intelligence' in people. We can call this the 'alien intelligence hypothesis.' It is entirely possible that the alien intelligence hypothesis is false. If the complex bag of phenomena we call 'intelligence' is something people use to interact with each other in human societies, an alien intelligence which didn't interact in the same way might not be seen by us as intelligence. And supposing alien intelligence did exist, could we recognise it *without* appealing to our human recognition of intelligence? Perhaps systems are just seen as intelligent in proportion to how well we can understand their patterns of behaviour. On this principle, computers (along with lettuce and beer cans) could already be intelligent, we

just can't recognise them as such.

We can, of course, take McCarthy's stance: "this is artificial intelligence and so we don't care if it's psychologically real" (Kolata, 1982). But as soon as we talk about minds we are talking about something psychological, so to compare minds and computers will inevitably be partly a psychological question. The actual nature of the distinction between human intelligence and artificial intelligence does matter. We—the people who are designing and evaluating these machines—are people with relatively uniform cultures, societies, and biologies—at least when compared to machines. Perhaps, as Searle (1992) claims, these factors affect human mental phenomena. If so, it would be surprising if they didn't also affect our recognition and interpretation of those phenomena.

The problem is this persistent anthropocentricity — we can't step outside our humanity although we perpetually see things as if they are independent of us. For physics that doesn't usually matter, but for psychological concepts such as 'intelligence,' we must remember that we are human. We need to discover what it is to be human before we can truly know where the differences between people and machines are.

2 Models for naive psychology

In looking at what is going on our heads when we see people as minds rather than as bodies, some of the most useful tools are models of the process of ascribing mental states to other systems. In this section, three candidate models will be examined in a little more detail, anthropomorphism, the simulation model, and the theory model.

1. Anthropomorphism. One way of ascribing mental states to a system is just to anthropomorphise it—to ascribe it human mental characteristics without reference to their real competences, but anthropomorphism is a complex and subtle phenomenon (Eddy et al., 1993) and not one that has been studied much. Eddy et al. (1993) looked at people's tendency to anthropomorphise animals, and suggest that there are two primary mechanisms involved: "people are likely to attribute similar experiences and cognitive abilities to other animals based on (1) the degree of physical similarity between themselves and the species in question (e.g. primates,) and (2) the degree to which they have formed an attachment bond with a particular animal (e.g. dogs and cats)" (Eddy et al., 1993).

Computers don't score too well on physical similarity, so this is likely to form a persistent bias against people ascribing mental states to them, unless we build them with a physical resemblance to us. Familiarity, fortunately, offers us a way out of this trap—we can in principle learn to see computers as minds.

There are several possible theories of anthropomorphism. Caporael (1986) suggests that it is a "'default schema' applied to non-social objects, one that is abandoned or modified in the face of contradictory evidence," but the evidence is against either animals or computers really being 'non-social' and familiarity can increase rather than decrease the tendency to anthropomorphise (Eddy et al., 1993). Alternatively, perhaps our tendency to anthropomorphise is really a disposition to take the "intentional stance" (Dennett, 1971), to see others as minds rather than as bodies. If, instead of taking the intentional stance, the physical stance is taken, the very different faculty of naive physics will be deployed. Anthropomorphism, then, determines whether or not an intentional stance will be taken, but it is not truly part of the stance itself. It plays the role of the rationality assumption in Dennett's model—although clearly anthropomorphism isn't the same thing as rationality—and the suggestion that the rationality assumption is "pre-theoretic" (Dennett, 1971) does allow us to interpret it as psychological rather than philosophical.

2. Simulation. Sometimes prediction of other people's mental states is better modelled by 'simulating' the other person, by pretending to *be* them, and to look at the world from their point of view. Clark suggests that a similar simulation process could even account for naive physics—perhaps Hayes' paper on liquids could be recast as a kind of simulation, and as far as the predictions are concerned, viewed externally, there needn't be any difference. For naive psychology, there is evidence that for some predictions—particularly those involving affective states—an ability to simulate other people works well (Hobson, 1993, Perner, 1991). Representational artificial intelli-

gence does simulation all the time—it's just another kind of hypothetical reasoning. Simulation, or taking another person's role, is a way that we can understand some aspects of another's mental states; for instance, to recognise somebody's ignorance.

So simulation is another way that we can reason about another's mental states. It works rather better for affective than for cognitive states (Hobson, 1993) but doesn't deal with everything: there are some tasks which children actually answer differently, but which they ought to answer the same if they use simulation to get the answer. Something is left over, and that something is a 'theory' of mind—not a theory in the scientific sense (Clark, 1987, Searle, 1992)—simply a theory in the sense of a set of tools for thinking about the unobservables of another person's mental states.

3. Theory. This theory aspect of prediction is that aspect which is most similar to the representational artificial intelligence. Some (e.g. Fodor) even take it as the complete answer to naive psychology, but this stretches it too far; a strong representational theory of mind is subject to too many philosophical and evolutionary objections (Dennett, 1987), and fails to account for all phenomena (Hobson, 1993). But just because a representational theory of mind can't provide a complete naive psychology doesn't mean that it doesn't form part of a complete naive psychology. The theory theory, as it is currently interpreted in psychology, describes naive psychology as a set of rules for dealing with the unobservable mental states of others. Its best analogue in artificial intelligence, therefore, would be a body of laws and heuristics for guessing at one other's mental states.

In artificial intelligence, the best programs for playing games like chess (and games are often a good metaphor for human social interaction) use a subtle mixture of simulation (look ahead) and theory (heuristics) because neither alone is sufficient. In principle, of course, a heuristic theory can generate a simulation and in practice an actual system—such as a trained connectionist network—might show aspects of theory and simulation under different circumstances, just as electrons can behave like particles or like waves.

These three models—anthropomorphism, simulation, and theory—represent different aspects of naive psychology rather than the whole, but they can be combined to create a complex composite model. When trying to predict or reason about the behaviour of a system, a complex of dispositions, one of which is anthropomorphism, selects a stance with respect to that system. These stances deploy natural faculties—so when dealing with a physical system, naive physics is applied, but for a psychological system, naive psychology is applied. Often, both stances could in principle be taken to the same system (even a thermostat, and although in practice there seems to be a mutual exclusion between the different stances (Dennett, 1971) this is where individual differences in the dispositions and the social context can influence how different people see the same system.

A mind will only be seen in the system from the intentional stance (Dennett, 1971)—that selected by anthropomorphism—and within the intentional stance as a whole there are different substances which depend on the access to the other's mental states that is required. If we are to 'simulate' it—to see what it is like to *be* the system—that can only happen if the system is believed to have the right kind of mechanism. The theory stance, on the other hand, is better at dealing with external, behavioural, questions.

How well do these models do? Although they barely hint at the true complexity of naive psychology, they do have some predictive power—one instance of this is in Woolgar's (1985) description of a device which bolts on to a video recorder and splices out advertisements during recording. On one level this is clearly intelligent behaviour, but if you then read the instructions, and they tell you that it actually works by detecting a particular signal in the transmission, this changes the ascription of intelligence, and "redefines and thus reserves the attribute of 'intelligence' for some future assessment of performance" (Woolgar, 1985). The change in our knowledge affects the stance that we take—affects whether or not we see the system from the intentional stance.

This integrated model shows a sensitivity to physical form and our knowledge of the system's design which is perhaps rather distressing for strong artificial intelligence. It seems to show not that it is impossible in principle, but that it is

just very hard for people to see things which don't physically, structurally, and behaviourally resemble people as being intelligent. Perhaps Brooks and Stein (1993) were right to design Cog with a humanoid form, not for any technical reason, but simply because it will make it easier for us to see Cog as an intelligent system.

3 Conclusions

Hayes concluded his "Second naive physics manifesto" (1985b) with a discussion on the importance of common sense for artificial intelligence. The reasons for this proposal are orthogonal to his, so the justifications are different.

Of all the naive disciplines proposed by Hayes and others, naive psychology is the only one that is obviously specifically human, but in all this work there is an implicit anthropocentricity. Right back to McCarthy's 1959 proposal of common sense, it was assumed that the common sense to be used is human common sense.

It is entirely possible that intelligent behaviour is distinguished not by an objective criterion of success, rationality, adaptiveness, or what have you, but by a subjective criterion of compatibility with our human naive psychology. At the core there was a simple problem: we forgot about anthropocentricity and took too much of what we intuitively felt to be right as being the truth. Stepping outside our humanity is something that perhaps we can never do in principle, but that doesn't mean that we shouldn't try—not by a regress to the Skinnerian vantage point (with apologies to Dennett, 1987) denying human mentalistic terms completely, but by indirectly looking at the effects of the ultimate unobservable, our anthropocentric point of view.

There is no strong methodological component to this proposal, because the project is just too important to be dismissed as a project merely on methodological grounds—and the same goes for naive physics. Dreyfus and Dreyfus claim, for example, that "the problem of finding a *theory* of common sense physics is insoluble because the domain has no theoretical structure" (Dreyfus & Dreyfus, 1988, original emphasis). This depends on what you want from the theory. Even if naive physics can't be described fully by reference to "abstract laws," that doesn't mean that we should give up. In the real world, theories aren't just right or wrong, but provide a greater or lesser measure of predictive competence—and even a partially correct theory is better than none.

This proposal is, like Hayes', a descriptive one: the construction of broad models of naive psychology. At least to start with, a broad and shallow approach is needed to sketch out naive psychology; it is not yet anywhere near as clearly structured into topics as Hayes presents naive physics. Pushing hard on one topic, like an air bubble under the wallpaper, might just move the problems somewhere else.

The problems that artificial intelligence is tackling are big ones—big enough to make some think that there are fundamental and possibly irretrievable flaws either in the discipline or even in the whole of science. This is an over-reaction; certainly our anthropocentricity is a big problem, but not one that is inaccessible in principle to science.

At the end of the day, we can all recognise intelligent behaviour when we see it. When we see people, we see them as minds, not just as bodies. When we see computers, we don't see minds. The difference between people and computers lies in ourselves as well as in them, and if we are to overcome this fundamental anthropocentric asymmetry, artificial intelligence must join up with psychology at least the extent of finding when and how we see minds. It must begin to study naive psychology.

References

[Brooks and Stein, 1993] Brooks, R. A. & Stein, L. A. (1993). Building brains for bodies. AI Memo 1439, MIT AI Laboratory.

[Byrne and Whiten, 1988] Byrne, R. W. & Whiten, A., editors (1988). *Machiavellian Intelligence: Social Expertise and the Evolution of Intellect in Monkeys, Apes, and Humans.* Oxford University Press, Oxford.

[Caporael, 1986] Caporael, L. R. (1986). Anthropomorphism and mechanomorphism: Two faces of the human machine. *Computers in Human Behavior*, 2(3):215–234.

[Clark, 1987] Clark, A. (1987). From folk psychology to naive psychology. *Cognitive Science*, 11:139–154.

[Cohen and Levesque, 1990] Cohen, P. R. & Levesque, H. J. (1990). Intention is choice with commitment. *Artificial Intelligence*, 42:213–261.

[Dennett, 1971] Dennett, D. C. (1971). Intentional systems. *Journal of Philosophy*, 68:87–106.

[Dennett, 1987] Dennett, D. C. (1987). *The Intentional Stance.* MIT Press, Cambridge, Massachusetts.

[Dreyfus and Dreyfus, 1988] Dreyfus, H. L. & Dreyfus, S. E. (1988). Making a mind versus modelling the brain: Artificial intelligence back at a branchpoint. *Daedalus*, 117(1):185–197.

[Eddy et al., 1993] Eddy, T. J., Gallup, G. G., & Povinelli, D. J. (1993). Attribution of cognitive states to animals: Anthropomorphism in comparitive perspective. *Journal of Social Issues*, 49(1):87–101.

[Hayes, 1985] Hayes, P. J. (1985). The second naive physics manifesto. In Hobbs, J. R. and Moore, R. C., editors, *Formal Theories of the Commonsense World*, pages 1–36. Ablex, Norwood, New Jersey.

[Hobson, 1993] Hobson, P. (1993). Understanding persons: The role of affect. In Baron-Cohen, S., Tager-Flusberg, H., & Cohen, D. J., editors, *Understanding Other Minds: Perspectives From Autism*, pages 204–227. Oxford University Press, Oxford.

[Humphrey, 1984] Humphrey, N. K. (1984). *Consciousness Regained.* Oxford University Press.

[Kolata, 1982] Kolata, G. (1982). How can computers get common sense? *Science*, 217:1237–1238.

[McDermott, 1987] McDermott, D. (1987). A critique of pure reason. *Computational Intelligence*, 3:151–160.

[Perner, 1991] Perner, J. (1991). *Understanding the Representational Mind.* MIT Press, Cambridge, Massachusetts.

[Searle, 1992] Searle, J. R. (1992). *The Rediscovery of the Mind.* MIT Press, Cambridge, Massachusetts.

[Woolgar, 1985] Woolgar, S. (1985). Why not a sociology of machines? the case of sociology and artificial intelligence. *Sociology*, 19:557–572.

Stuffing Mind into Computer: Knowledge and Learning for Intelligent Systems

Kevin J. Cherkauer
Department of Computer Sciences
University of Wisconsin-Madison
1210 West Dayton St., Madison, WI 53706, USA
Phone: 1-608-262-6613, Fax: 1-608-262-9777
E-mail: cherkauer@cs.wisc.edu
http://www.cs.wisc.edu/~cherkaue/cherkauer.html

Keywords: artificial intelligence, knowledge acquisition, knowledge representation, knowledge refinement, machine learning, psychological plausibility, philosophies of mind, research directions

Edited by: Marcin Paprzycki
Received: May 10, 1995 **Revised:** November 21, 1995 **Accepted:** November 28, 1995

The task of somehow putting mind *into a computer is one that has been pursued by artificial intelligence researchers for decades, and though we are getting closer, we have not caught it yet. Mind is an incredibly complex and poorly understood thing, but we should not let this stop us from continuing to strive toward the goal of intelligent computers. Two issues that are essential to this endeavor are knowledge and learning. These form the basis of human intelligence, and most people believe they are fundamental to achieving similar intelligence in computers. This paper explores issues surrounding knowledge acquisition and learning in intelligent artificial systems in light of both current philosophies of mind and the present state of artificial intelligence research. Its scope ranges from the mundane to the (almost) outlandish, with the goal of stimulating serious thought about where we are, where we would like to go, and how to get there in our attempts to render an intelligence in silicon.*

1 Introduction

The ultimate goal of artificial intelligence (AI) is to somehow implement a very wonderful and complex thing we call "mind" within the confines of an artificial computer. Even if undaunted by the incredible paucity of our own understanding of mind, we may nonetheless find ourselves put off by the sheer complexity and size we usually imagine this machinery must entail. Despite our inability to satisfactorily define intelligence, one component we generally feel must be present is a large store of *knowledge* about every aspect of the world. However, it helps us little to decide, "Let us put everything we know into a computer." How do we represent this knowledge? How do we refine it? And how do we get it into the system? Surely we do not have time to put *everything* in by hand!

Perhaps our systems can use learning to acquire and modify the knowledge they need largely on their own. Instead of trying to stuff our own brains into the computer one bit at a time (Figure 1), perhaps we can write programs that let the computers learn for themselves what they need to know. Learning is, after all, the way humans fill their own brains with knowledge. But how much can we gain from human analogies? Is psychological plausibility a necessity or a curse? Will our machines need emotional motivation in order to be truly successful learners? The questions, as always, come thick and fast.

In this paper we will take a moment to examine these issues of knowledge and learning in the light of both current philosophies of mind and the present state of artificial intelligence research. It is not often, in the world of technical papers, that we allow our thought processes to roam free. That is

Figure 1:

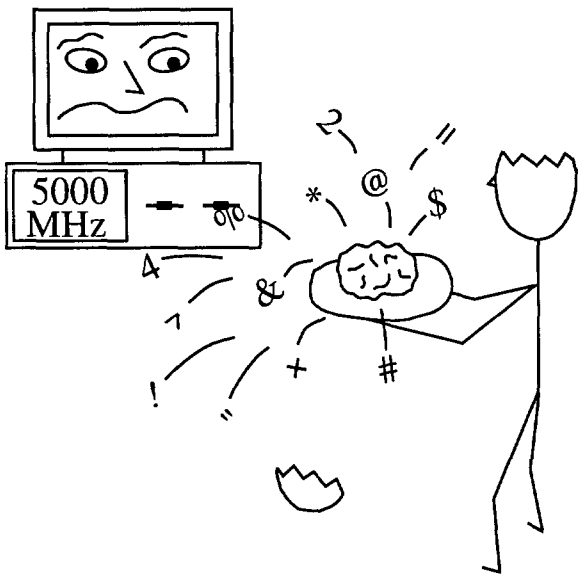
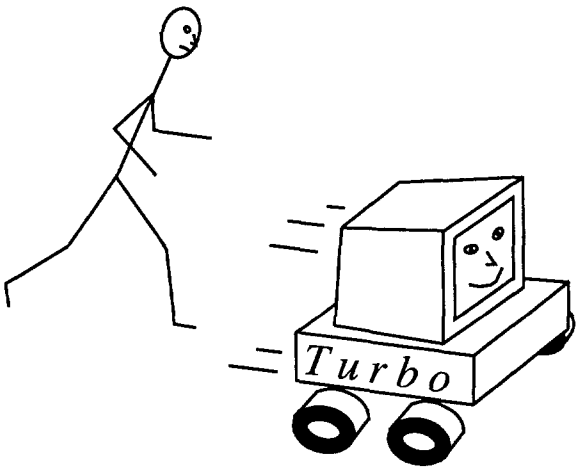


Figure 2:



the main goal of this paper—to visit some of the wild pastures of imagination that spawned the field of AI in the first place. We hear these days that all those far-flung dreams of intelligent computers from decades ago are still as out of reach as ever. We spend too much of our time being apologetic, trying to present AI advances in as narrow a scope as possible, almost as if we wish them to appear insignificant in order to avoid accusations of chasing hopeless fantasies. It is indeed important to keep a firm grip on reality—I do not think anyone would argue otherwise. But if we are truly to achieve wonders, we must first allow ourselves to imagine them. I hope you will join me in doing so!

2 How Should Our Systems Acquire Knowledge?

The question of how to get knowledge into our systems is a key issue in building intelligences. Most expert systems currently acquire knowledge through painstaking hand programming by a knowledge engineer working closely with a domain expert. A major goal of AI is to produce machines that perform intelligent tasks, so a dedicated AI researcher may suggest that the best answer to our question, "How should our systems acquire knowledge?" is, "Why, through machine learning, of course!" Some obvious advantages of automating the knowledge acquisition process through machine learning (ML) are speed and accuracy of rule construction. However, to succeed in this endeavor we must somehow develop ML techniques which are as good at creating sets of rules for specific domains as an expert human knowledge engineer. This just pushes the problem of emulating expert behavior one level deeper: in trying to avoid hand coding a program that embodies the knowledge of a domain expert, we find we must now hand code a program that embodies that of a knowledge engineer!

We may still manage to tackle this problem if we can find some way to make the knowledge engineer's knowledge easier to program than the domain expert's. Humans use their knowledge and intelligence to construct expert system knowledge bases. Our comparatively dim-witted computers' only chance to overcome their own lack of insight is their blinding speed and tireless persistence (Figure 2) and their utter disdain for the human propensities toward fatigue, boredom, distraction, careless mistakes, and other such egregious vices. Since these are the computer's fortes, we must exploit them.

For instance, we can have our machines search very large numbers of possible rules and rule fragments to find a good set. Whereas a human knowledge engineer examines only a few alternative rules, banking on the domain expert's deep understanding of the problem to insure a good solution, the less knowledgeable computer must succeed through perseverance. The FOIL system [38] is one example of a large-scale search appro-

ach to construct predicate calculus rules describing a domain. Part of my own recent work [8, 9] has concentrated on high-speed parallel search methods to sift through hundreds of thousands of potentially useful features for representations that make learning easier.

Computers have an advantage over people in dealing with huge volumes of data. In many cases a problem is too complex and poorly understood for people to construct effective rules to solve it. All that is really available is a large set of raw data. Object recognition, image understanding, speech production, argument construction, complex motor skills, breast cancer prognosis, and protein folding prediction are all real-world problems that fit this description. Some of these are problems of perception and action that humans accomplish effortlessly, yet we cannot articulate how we do so. Others are more abstract problems of interest to science and medicine. All of them have been the subject of machine learning research (e.g. [7, 25, 35, 37, 40, 44, 45, 46]).

This is not to say we should require our machines to learn absolutely everything from scratch. We should certainly take advantage of existing domain knowledge, both low- and high-level, to the extent we can afford it. There is no reason to learn logical inference rules from first principles when we can easily code them into a knowledge base. Likewise, if a domain expert can provide partial sets of high-level rules or other advice, this will jump-start the system and reduce the amount learning time and data required [21, 34, 50]. Guidance from domain knowledge may also be crucial to prevent so-called "oversearching" [39], or the discovery of spurious correlations during learning. Unfortunately, human expertise is too expensive to allow us to hand code everything in a system of the size and complexity needed for intelligence. The builders of the monumental CYC knowledge system, though willing to invest large amounts of effort to hand code much of the knowledge, nonetheless advocated automating this process as much as possible through ML techniques even from the early stages [19], and they continued to add learning mechanisms over the years [17]. As the intelligent systems we design become increasingly sophisticated, we have no choice but to adopt machine learning techniques as facilitators. To reach human-level intelligence, an artificial system must be enormously more complex than anything we have created to date. The journey to machine intelligence will be shortest if we continue to develop and apply the powers of machine learning on this quest.

3 What Form Should the Knowledge Take?

A serious problem with using ML for knowledge acquisition is what Michalski terms the "knowledge ratification bottleneck" [23]. That is, for applications in which malfunction could have costly, critical, or even life-threatening consequences, any knowledge a system uses must be closely examined for correctness. It is difficult enough to do this with large knowledge bases written by humans; the problem is only compounded if they are cobbled together automatically by a machine. Michalski contends that in such situations, the explanation capabilities of ML systems must be well developed, and the knowledge representation used should be comprehensible to humans. These constraints seem to favor the symbolic, rule-like representations we have spoken of so far over other alternatives like connectionism.

Or do they? Are huge rule bases of the scale needed to simulate human-level intelligence any more comprehensible than artificial neural networks (ANNs)? On the other hand, why can not connectionist representations be made as understandable as rules? Mitchell and Thrun [25] develop ANNs which model various primitive robot actions and then treat these networks as if they were rules. Others have developed methods that allow the extraction of symbolic rules from trained neural networks [10, 13, 14, 42, 48, 49], so the two representation styles are not as irreconcilable as they look.

The question of what form knowledge should be *stored* in relates to the question discussed in the previous section of how a system *acquires* knowledge. If learning is used to do this, many different internal representations are possible, rules and ANNs among them. The hand coding approach, in which humans construct the knowledge base, generally favors a symbolic storage representation. However, there exist machine learning systems that can store and refine initially symbolic knowledge in connectionist ANNs (see Section 5),

so there is no reason hand-coded knowledge must remain in its original form.

There are arguments other than understandability for preferring symbolic knowledge structures. Higher-level human cognitive processes operate in an apparently symbolic fashion, perhaps suggesting we should use similar approaches in computers. However, a connectionist might reply that the perceived symbolic nature of our reasoning processes is an illusion, as the brain is a connectionist device. A third person might dismiss both of these arguments, claiming it does not matter how *humans* solve problems if our goal is to build *machines* to do the same. The classic conflicts over psychological and physiological plausibility persist. Let us explore these conflicts further in the next section.

4 Psychological Plausibility: Friend or Foe?

A common argument against using rules to describe knowledge is that of psychological (and sometimes physiological) plausibility. The brain is physically a connectionist device. It is tempting thus to equate psychological plausibility with connectionist implementations, but in fact it is less clear how much the details of abstract cognition depend directly on the connectionist nature of the hardware. It is not unknown for discussions of these questions to become quite animated, especially as there seem to be almost as many points of view as there are interested parties. An imaginary conversation may help us to better understand the extent of the rifts that exist.[1]

Engineer: Psychological plausibility is just a meaningless hoop to jump through, completely superfluous to our goal of building thinking machines! It's hard enough to get anything like intelligence out of a computer even without a bunch of arbitrary anthropocentric constraints. Now you're telling me you won't be satisfied with mere human-*like* intelligence, but you insist on human-*structured* intelligence to boot! Next you'll demand android bodies, vat-grown neural brains, and probably even—*emotions!* We should just go with what works regardless of what it looks like.

Psychologist: How can you take such a position when the human mind is our only example of advanced intelligence? Only incredible arrogance would let us imagine we can start from scratch, ignoring everything psychology has to tell us, and do a better job. If we ever want our systems to speak to us as peers, they will have to understand things the same way we do. It is sheer folly to attempt a computer intelligence that conflicts with our accumulated body of psychological knowledge.

Neurobiologist: *(Clapping hands.) Bravo!* But the psychologist does not go far enough. I'll grant that we know a few things about human cognition, but we have even more specific knowledge about the hardware that implements it. We know exactly how neurons fire, what chemicals they use to transmit signals across synapses—even their patterns of connection in some parts of the brain. AI's best bet is to simulate this hardware as closely as possible, as it is the only thing we thing we have a concrete description of.

Engineer: Ah ha! *(Dons a smug look.)* I *knew* someone would want vat-grown brains!

Philosopher: *(With a sly look.)* Hold on! Why are we limiting our vision to puny, human-like machine intelligences? Shouldn't our goal be to create machines that are *smarter* than people? We can't copy knowledge from adults to babies or put people through a thousand years of education, but we can build computer memories big enough to hold entire libraries and processors fast enough to digest them. Does piscine plausibility help us build nuclear submarines? Does avian plausibility help us build airliners? *(Throws up hands.)* Absolutely not! In fact, these things merely hold us back!

Indeed, there seem to be two diametrically opposed and largely antagonistic camps with respect to this issue: those who believe that psychological (or even biological) plausibility is essential to producing an intelligent artificial system, and those who believe these requirements are merely contrived obstacles that slow our progress or limit

[1] Of course, there are many more points of view within a given field than these caricatures present.

the goals we set for AI. One is tempted to say that what we need most of all is a moderate voice, a compromiser, a fence-sitter—perhaps even a

Politician: Ah, you people are hopeless. The problem is hard enough without all these religious schisms. We should use what ideas we can from psychology without promising to produce a psychologically plausible computer system. We should look to neurobiology for insight without promising vat-grown brains—or even neural networks. We should apply machine learning without promising that every component of the final system will be automatically generated instead of hand programmed. We should follow visions from philosophy without promising to realize them without revision (if I may be so bold as to pun). In short *(waving hands)*, we should take everything we can get our hands on and guarantee nothing in return! (Er...that didn't come out quite right....)

Underlying all this waffling is an important issue which has so far remained implicit, and that is the distinction between the hardware on which an algorithm is implemented and the algorithm itself. Von Neumann [27] states unequivocally that, while we understand the abstract concepts of logic and mathematics in a symbolic way, these concepts must necessarily be implemented very differently in human brains than in digital computers because of fundamental hardware differences. The brain is a massively parallel, low-precision device that encodes information robustly via statistical patterns and performs relatively short chains of calculation. Digital computers are (much more) serial and depend on long chains of brittle, high-precision calculations in which a single corrupted bit can cause a system crash. When we speak of logic and mathematics, we are really using a pseudocode that describes the algorithm without saying anything about the details of implementation. Symbolic descriptions of high-level natural languages and reasoning systems tell us little about their biological implementations. The implication is that they will tell us no more about how to implement them in digital computers.

For these reasons, I believe the most fruitful approach to resolving the controversy of this section is to view psychology and biology as tools for discovering the *algorithms* the human brain runs. Knowing the algorithms, we can then focus on producing the (radically different) *implementations* required for digital computers where this appears suitable. We must keep in mind that the brain's massively parallel algorithms may often be impractical under serial reimplementation due to time or space requirements [43]. There will also be many cases where psychological and biological study are unable to glean the specific algorithm the brain uses to solve a given problem. In these situations, we must resort to more bottom-up engineering that takes best advantage of the strengths of digital computers to arrive at alternate solutions. One example of success using this approach is that of chess playing programs. Although few would argue that human grand masters and computers implement the same chess playing algorithms, it is impossible to deny that computers can play chess at the grand master level. In this problem, an alternate algorithm based on high-speed serial search has achieved the same quality of results as the very different process of high-level human reasoning.

To summarize, psychology and biology should be treated as two tools among many the AI researcher can use to gain insight into methods of intelligent problem solving, but they should not be seen as the only legitimate tools in the arsenal. While the computational properties of the brain and digital computers do overlap, they are far from identical. We can gain algorithmic insight from the brain's solutions, but we will certainly need to tailor these solutions, and often radically alter them, to fit the differing properties of the computer. I do not think there is much to gain by demanding psychological plausibility, whatever that may be, in computer systems that are by nature so unlike the brain, nor do I think there is any real justification in this context to prefer so-called "connectionist" over "symbolic" computer implementations or vice versa.[2] Our time is better spent developing and testing algorithms than arguing about these points.

[2] Comprehensibility of the knowledge base, which favors symbolic representations, is a separate issue.

5 Knowledge Refinement

As research continues on the problem of using ML for knowledge acquisition, we will develop more guided approaches than the weak search methods. One step that has already been taken in this direction is that of automatically refining incorrect or partial domain knowledge [4, 11, 12, 15, 16, 20, 21, 22, 26, 30, 31, 32, 33, 34, 47, 50]. Even if we do not have a fully satisfactory set of rules for solving a problem, our learning algorithms can still benefit from the incomplete knowledge we do have. Knowledge refinement systems such as those cited are often able to use partial knowledge to produce better solutions to real-world problems than was previously possible with weak methods alone.

Knowledge refinement systems, like other learning systems, can be symbolic or connectionist. A symbolic approach typically starts with a set of imperfect rules from a human expert and iteratively modifies it in order to improve its correctness or coverage, e.g. by adding and deleting terms. EITHER [32, 33] and NEITHER [4] are systems which refine propositional Horn clause rule sets in such a manner, and FORTE [26] extends the technique to function-free Horn clause representations of logic programs.

The KBANN family of algorithms represents a connectionist knowledge refinement approach. It translates a set of propositional rules [50] or a description of a finite state automaton [20] into the nodes and weights of an ANN. The network, and therefore its embodied knowledge, is then refined by standard ANN backpropagation training [41]. One can then either use the modified network as it stands or apply methods to extract symbolic rules from it [10, 13, 14, 42, 48, 49].

Knowledge refinement systems can take advantage of partial knowledge and correct and embellish it automatically through ML techniques. Their use will greatly reduce the effort needed to create knowledge bases for intelligent systems.

6 Are Rules Sufficient?

There is a possibility that some problems simply cannot be solved by symbolic rules. Perhaps the reason human cognitive processes are so hard to pin down is that they operate in a fundamentally distributed and unrule-like way. Chaos theory tells us the only accurate model of the weather is the weather itself. The idea that the world is its own best model has sometimes been used to argue against knowledge representation in any form [2, 3, 5, 6]. Perhaps the only way to model human cognition is through a device that is similar in structure and complexity to the human brain [27]. Penrose [36] suspects that the physics of brain operation makes some of our thought processes (especially the feeling of awareness) nonalgorithmic, questioning the "strong AI" position that all our thinking is merely the enacting of some algorithm. If this is true, we may have no hope of modeling these aspects at all, either by symbolism or connectionism, using current computer architectures.

I would like to challenge the extremity of these positions. Though it is true that we cannot precisely model the weather at a micro scale, this does not mean there is no high-level structure amenable to abstraction. A meteorologist does not need to predict the temperature of every cubic centimeter of air to tell us it will drop when a cold front moves in. This is a simple symbolic rule with real predictive power.

In chaotic domains, any model at all—symbolic or not—must approach the complexity of the system itself in order to achieve arbitrary accuracy, but this misses the point of having a model in the first place. One needs only a very small set of rules to do better than chance in predicting the next day's weather. One of the simplest and most accurate systems for one-day weather forecasting consists of a single rule: "Tomorrow's weather will be the same as today's." Simplification through models allows us to find order and understanding where there would otherwise be none.

In this vein, symbolic rules may be used to model the processes of cognition, even though the brain's implementation is a distributed one. Much of our thinking can be described symbolically. We communicate with one another with symbols, and we store knowledge in external libraries and other media in the same way. There is thus plenty of reason to expect rule-driven symbol manipulation à la the classic *Physical Symbol System Hypothesis* [29] to be a reasonable model for many aspects of human intelligence. Just as we need not reproduce every detail of bird anatomy to make an airplane that flies, we need not

reproduce every cell and connection of the brain to make a machine that thinks. I believe symbolic rules are sufficient to capture most aspects of human intelligence at the everyday level of granularity most useful to us, even though at a micro level they will operate differently than the human brain.

7 Are Emotions Necessary for Learning?

Whether we need to include emotions in our learning systems may seem like a strange question, but with a moment's thought we realize that much of human learning is motivated by emotions. Our engineer of Section 4 spoke of emotions as if they were totally irrelevant to machine intelligence. However, the same cannot be said of human intelligence. Children must receive love and nurture to survive and thrive. Emotional involvement is a powerful motivator in their development and success and continues to be throughout adulthood. In a classic essay, Hadamard [18] investigates the role of human emotion in fostering creative discovery and invention. If we hope to build truly intelligent machines, might we not also need to build in such a motivating drive? Even if it is not completely necessary for artificial systems, can we afford to ignore this complex and powerful urge to learn?

In *The Society of Mind* [24], Minsky casts emotions as fundamental to the success of our intelligence. They spur our creativity while preventing us from obsessively fixating on a single idea or purpose. Without them, we would become robotic drones and accomplish little. Emotions are important checks and balances in the complex system of mind. However, Minsky does not attribute any special status to emotions. He views them simply as tools that interacting mental agents use to accomplish their goals. For example, he describes *Anger* as a tool agent *Work* can exploit to prevent agent *Sleep* from gaining control of the mind. No mysterious qualities need be assigned to *Anger* to explain it. It is simply one of many competing mechanisms which help get things done in the mind.

Newell [28], on the other hand, defines intelligence without reference to emotions. For Newell, intelligence depends only on how well a system uses the knowledge it has. Perfect use constitutes perfect intelligence, while a system that ignores its knowledge has no intelligence.

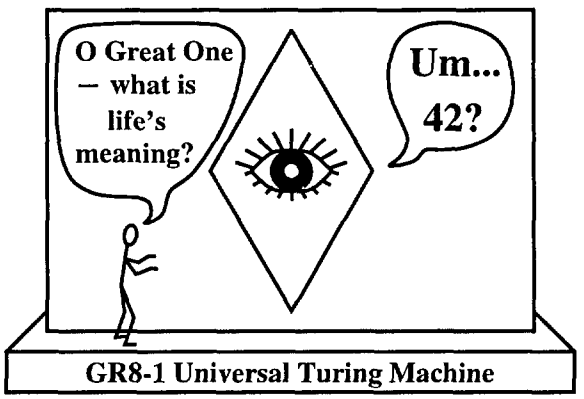
I find Newell's definition flawed specifically where emotions and learning are concerned. A system with emotions may have a curiosity that leads it to formulate and test theories about the world. It does not know whether these theories are true, nor does it know it may benefit from testing them, so any exploration and learning arising from this curiosity do not count in Newell's definition of intelligence. An otherwise identical system that lacks the motivation of curiosity, and so learns nothing, is considered equally intelligent. Nonetheless, empirical investigation often leads to new knowledge that can improve life for the system. Would we not credit a curious, exploring, experimenting system that continually expands its own knowledge base, capabilities, and efficiency (and happiness, perhaps?) with more intelligence than a mentally sedentary one that mechanically applies the same old knowledge to just get by? I hope we would!

Does this imply that emotions are *necessary* for learning? Not at all. While emotions play a key role in motivating human learning, they are certainly not the only possible incentives for learning in general. One may sharpen a skill simply by repeating a task many times, whether one intends to become better at it or not. One may make a great discovery purely by accident. Furthermore, computers are not humans, and they can be motivated in other ways. In a computer system, learning may simply be something the machine is required to do by its program. Both emotions and learning should be important components of any definition of intelligence, but emotions are not prerequisite for learning to occur.

8 Building Superintelligences

Most of the time the ultimate goal of AI is stated as building an artificial intelligence of human capabilities, as suggested by the famous *Turing test* [51]. As long as we are being ambitious, however, why not aim for intelligences that are even greater? Why stop with a machine Albert Einstein if we can hope for even more? Even though this is far beyond our present capabilities, it should still be a subject to think about (Figure 3).

Figure 3:



Assuming we had already reached the goal of creating machines as smart as individual humans, what would be our next step toward the higher goal of superintelligences? One avenue to explore is that of societies of intelligent agents. We could seek emergent superintelligence from the interactions of "regular" intelligences in much the same way Minsky seeks emergent intelligence from the interactions of unintelligent agents in *The Society of Mind* [24]. This may be a useful insight, but we must examine it more closely to reap its potential benefits. To wit, if our Einstein unit (person or machine) has an IQ of 300, do three average people (100 IQ each) equal one Einstein? I doubt it. They are probably more like 0.4 Einsteins. One might therefore argue that we just need ten or so average people to boost up to one Einstein. I don't buy this either—there is surely a law of diminishing returns operating such that each successive person adds progressively less to the Einstein index, even if only due to communications problems.

Does a colony of thousands of micro-Einsteinian ants ever approach an Einstein of intelligence? Probably not, but ants may be a bad example—their interagent communication and knowledge storage capabilities are surely quite limited. Perhaps the only real problem with applying the extended society of mind metaphor to humans is that humans are too loosely coupled (i.e. our communication bandwidth is too low). We can store as much knowledge as we want using external media. The problem is only in how quickly we can process and apply it.

I postulate that sophisticated symbolic communication among intelligent agents is sufficient to achieve emergent superintelligence. The main reason we do not see obvious mega-Einsteinian strides in the intelligence of cooperating groups of people is slow communication.

If low bandwidth is the only substantive obstacle in the path of emergent superintelligence in human societies, we should be able to see evidence of mega-Einsteinian accomplishments if we observe societies for a long enough period of time. And lo—this is exactly what we *do* see in the rise of technological societies! The knowledge and achievements of these systems is vastly greater than anything a single human could ever accomplish, no matter how smart. So without even realizing it, we already have hard evidence of the success of this approach to building superintelligences! If we could implement in a machine (or machines) a large number of intelligent agents communicating and interacting mentally at high speeds, we might get somewhere in our fantasy project of producing a time-localized, mega-Einsteinian reasoner.

Since humans will probably not be networking their minds together telepathically any time soon, our best hope for a high-speed, superintelligent reasoning system is to build an artificial one. Interacting conglomerations of intelligent agents present a realistic paradigm for achieving this. In the mean time, we should reexamine the idea of human-level intelligence emerging from collections of interacting unintelligent agents. I believe this is the most likely route to our first truly intelligent machine.

9 Conclusion

We have explored some important current issues of knowledge and learning for the creation of artificial intelligence, raising many questions and, hopefully, a few answers in the process. If my presentation has also raised a few eyebrows, so much the better. I believe that knowledge and learning are both essential to the enormous task of implementing intelligent artificial systems, and research on these fronts is steadily progressing. At the same time, as we toil through the technical details of basic research, we should not lose the ability to dream of greater things for tomorrow. It is these dreams that will make intelligent machines a reality.

10 Acknowledgments

Like most human intellectual achievements, this paper is the product of many brains. I would like to thank those who provoked my thoughts on these subjects through symbolic communication, either spoken or written, especially Larry Travis, who inspired the character of the Philosopher in Section 4; Derek Zahn, who exposed me to different points of view; Marvin Minsky, whose ideas [24] helped mine to germinate; three anonymous reviewers, whose comments and suggestions greatly improved the paper; and, of course, Douglas Adams, whose work [1] elicited Figure 3.

References

[1] Douglas Adams. *The Hitchhiker's Guide to the Galaxy.* Harmony Books, New York, NY, 1980.

[2] P.E. Agre and D. Chapman. Pengi: An implementation of a theory of activity. In *Proc. 6th Nat'l Conf. on Artificial Intelligence*, pages 268–272, Seattle, WA, 1987.

[3] P.E. Agre and D. Chapman. What are plans for? *Robotics and Autonomous Systems*, 6:17–34, 1990.

[4] P.T. Baffes and R.J. Mooney. Symbolic revision of theories with M-of-N rules. In *Proc. 13th Int'l Joint Conf. on Artificial Intelligence*, pages 1135–1140, Chambéry, Savoie, France, 1993. Morgan Kaufmann.

[5] R.A. Brooks. Elephants don't play chess. *Robotics and Autonomous Systems*, 6:3–15, 1990.

[6] R.A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[7] M.C. Burl, U.M. Fayyad, P. Perona, P. Smyth, and M.P. Burl. Automating the hunt for volcanoes on Venus. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition: Proc.*, Seattle, WA, 1994. IEEE Computer Society Press.

[8] K.J. Cherkauer and J.W. Shavlik. Protein structure prediction: Selecting salient features from large candidate pools. In *Proc. 1st Int'l Conf. on Intelligent Systems for Molecular Biology*, pages 74–82, Bethesda, MD, 1993. AAAI Press.

[9] K.J. Cherkauer and J.W. Shavlik. Selecting salient features for machine learning from large candidate pools through parallel decision-tree construction. In H. Kitano and J.A. Hendler, editors, *Massively Parallel Artificial Intelligence*, pages 102–136. AAAI Press/MIT Press, Menlo Park, CA/Cambridge, MA, 1994.

[10] M.W. Craven and J.W. Shavlik. Using sampling and queries to extract rules from trained neural networks. In *Machine Learning: Proc. 11th Int'l Conf.*, pages 37–45, New Brunswick, NJ, 1994. Morgan Kaufmann.

[11] S.K. Donoho and L.A. Rendell. Rerepresenting and restructuring domain theories: A constructive induction approach. *Journal of Artificial Intelligence Research*, 2:411–446, 1995.

[12] L.M. Fu. Integration of neural heuristics into knowledge-based inference. *Connection Science*, 1(3):325–340, 1989.

[13] L.M. Fu. Rule learning by searching on adapted nets. In *Proc. 9th Nat'l Conf. on Artificial Intelligence*, pages 590–595, Anaheim, CA, 1991. AAAI Press.

[14] S.I. Gallant. *Neural Network Learning and Expert Systems.* MIT Press, Cambridge, MA, 1993.

[15] A. Ginsberg. *Automatic Refinement of Expert System Knowledge Bases.* Pitman, 1988.

[16] A. Ginsberg. Theory reduction, theory revision, and retranslation. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, pages 777–782, Boston, MA, 1990. AAAI Press/MIT Press.

[17] R.V. Guha and D.B. Lenat. Cyc: a midterm report. *AI Magazine*, 11(3):32–59, 1990.

[18] J. Hadamard. *An Essay on the Psychology of Invention in the Mathetmatical Field.* Princeton University Press, Princeton, NJ, 1945.

[19] D. Lenat, M. Prakash, and M. Shepherd. CYC: using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI Magazine*, 6(4):65–85, 1986.

[20] R. Maclin and J.W. Shavlik. Refining domain theories expressed as finite-state automata. In *Machine Learning: Proc. 8th Int'l Wkshp.*, pages 524–528, Evanston, IL, 1991. Morgan Kaufmann.

[21] R. Maclin and J.W. Shavlik. Creating advice-taking reinforcement learners. *Machine Learning*, 22(1–3), 1996.

[22] J.J. Mahoney and R.J. Mooney. Combining connectionist and symbolic learning to refine certainty-factor rule-bases. *Connection Science*, 5(3–4):339–364, 1993.

[23] R.S. Michalski. Understanding the nature of learning: Issues and research directions. In R.S. Michalski, J.G. Carbonell, and T.M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach, Volume II*, pages 3–25. Morgan Kaufmann, San Mateo, CA, 1986.

[24] M. Minsky. *The Society of Mind.* Simon and Schuster, New York, NY, 1986.

[25] T.M. Mitchell and S.B. Thrun. Explanation-based neural network learning for robot control. In *Advances in Neural Information Processing Systems*, volume 5, Denver, CO, 1993. Morgan Kaufmann.

[26] R.J. Mooney and B.L. Richards. Automated debugging of logic programs via theory revision. In *Proc. Second Intl. Wkshp. on Inductive Logic Programming*, Tokyo, Japan, 1992.

[27] J. von Neumann. *The Computer and the Brain.* Yale University Press, New Haven, CT, 1958.

[28] A. Newell. *Unified Theories of Cognition.* The William James Lectures, 1987. Harvard University Press, Cambridge, MA, 1990.

[29] A. Newell and H.A. Simon. Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3), 1976.

[30] D.W. Opitz and J.W. Shavlik. Heuristically expanding knowledge-based neural networks. In *Proc. 13th Int'l Joint Conf. on Artificial Intelligence*, volume 2, pages 1360–1365, Chambéry, Savoie, France, 1993. Morgan Kaufmann.

[31] D.W. Opitz and J.W. Shavlik. Using genetic search to refine knowledge-based neural networks. In *Machine Learning: Proc. 11th Int'l Conf.*, pages 208–216, New Brunswick, NJ, 1994. Morgan Kaufmann.

[32] D. Ourston and R.J. Mooney. Changing the rules: A comprehensive approach to theory refinement. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, pages 815–820, Boston, MA, 1990. AAAI Press/MIT Press.

[33] D. Ourston and R.J. Mooney. Theory refinement combining analytical and empirical methods. *Artificial Intelligence*, 66(2):273–309, 1994.

[34] M. Pazzani and D. Kibler. The utility of knowledge in inductive learning. *Machine Learning*, 9(1):57–94, 1992.

[35] E.P.D. Pednault. Some experiments in applying inductive inference principles to surface reconstruction. In *Proc. 11th Int'l Joint Conf. on Artificial Intelligence*, pages 1603–1609, Detroit, MI, 1989. Morgan Kaufmann.

[36] R. Penrose. On the physics and mathematics of thought. In R. Herken, editor, *The Universal Turing Machine: A Half-Century Survey*, pages 491–522. Oxford University Press, Oxford, England, 1988.

[37] D.A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural Computation*, 3:88–97, 1991.

[38] J.R. Quinlan. Learning logical definitions from relations. *Machine Learning*, 5:239–266, 1990.

[39] J.R. Quinlan and R.M. Cameron-Jones. Oversearching and layered search in empirical learning. In *Proc. 14th Int'l Joint Conf. on Artificial Intelligence*, volume 2, pages 1019–1024, Montréal, Québec, Canada, 1995. Morgan Kaufmann.

[40] B. Rost and C. Sander. Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*, 232:584–599, 1993.

[41] D.E. Rumelhart, G.E. Hinton, and R. Williams. Learning internal representations by error propagation. In D.E. Rumelhart and J.L. McClelland, editors, *Parallel Distributed Processing*, volume 1, pages 318–363. MIT Press, Cambridge, MA, 1986.

[42] K. Saito and R. Nakano. Medical diagnostic expert system based on PDP model. In *Proc. IEEE Int'l Conf. on Neural Networks*, pages 255–262, San Diego, CA, 1988. IEEE Press.

[43] H. Schnelle. Turing naturalized: Von neumann's unfinished project. In R. Herken, editor, *The Universal Turing Machine: A Half-Century Survey*, pages 539–559. Oxford University Press, Oxford, England, 1988.

[44] T.J. Sejnowski and C.R. Rosenberg. Parallel networks that learn to prounounce English text. *Complex Systems*, 1:145–168, 1987.

[45] D.B. Skalak and E.L. Rissland. Inductive learning in a mixed paradigm setting. In *Proc. 8th Nat'l Conf. on Artificial Intelligence*, Boston, MA, 1990. AAAI Press/MIT Press.

[46] W.N. Street, O.L. Mangasarian, and W.H. Wolberg. An inductive learning approach to prognostic prediction. In *Machine Learning: Proc. 12th Int'l Conf.*, pages 522–530, Tahoe City, CA, 1995. Morgan Kaufmann.

[47] K. Thompson, P. Langley, and W. Iba. Using background knowledge in concept formation. In *Machine Learning: Proc. 8th Int'l Wkshp.*, pages 554–558, Evanston, IL, 1991. Morgan Kaufmann.

[48] S.B. Thrun. Extracting provably correct rules from artificial neural networks. Technical Report IAI-TR-93-5, Institut für Informatik III, Universität Bonn, 1993.

[49] G.G. Towell and J.W. Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine Learning*, 13(1):71–101, 1993.

[50] G.G. Towell and J.W. Shavlik. Knowledge-based artificial neural networks. *Artificial Intelligence*, 70(1,2):119–165, 1994.

[51] A.M. Turing. Computing machinery and intelligence. *Mind*, 59:433–460, 1950.

unsuitable for inclusion in tier #3 all but sacrificed lexical resemblance for the sake of semantic content. Consider this imaginative offering, perhaps from a disgruntled concert-goer: "All mimmicing [sic] were the composers / And the public is promptly outraged."

This tier also fell prey, early on, to a dictionary which unfortunately contained the word "rath" ("a pre-historic hill-fort"). This unintended reward served to stimulate what I suppose to be another psychological tendency. Picture if you will the earnest human subject, dutifully looking up non-existent word after non-existent word, experiencing repeated disappointment if not frustration, and trying to puzzle out what, if anything, is intended by the text. Finally, on the penultimate word of the last line, the subject strikes paydirt. Contrary to conditioned negative expectation, an unknown word actually appears in the dictionary! In consequence, it becomes a focus of meaning for the entire line; e.g. "And the fairies dug out raths." This is another clever attempt to couch semantic content in lexical resemblance, and the subject even pencilled notes which illustrate her thought-processes: "mome" was interpreted as "gnome" (which was then pluralized and transmuted to "fairies"), while "outgrabe" conveniently became "dug out". Another subject was evidently so transported at finding "rath" in the dictionary that he substituted its definition wholesale: "And the wonderous [sic] hill-forts camouflaged". This conveys a palpable chunk of meaning, and it even harbours a vestige of resemblance. Although I speedily replaced the dictionary in question with a less fortified edition, I could not dispossess the human subjects of their penchants to make the stanza more grammatical and meaningful.

While the half-dozen most suitable versions offered sufficient ingenuity and variety to motivate tier #3, they also posed one serious problem. All the human subjects—whether aided by the dictionary or not—had identified and variously substituted for the contraction " 'Twas". The computer alone had come up with " 'Twos", and its version seemed obviously distinguishable from the others on that ground alone. So I wilfully engaged in a subterfuge. I contrived version (8) as a decoy, which not only reproduced the problematic distinguishing feature of (1), but which also did things that I thought the computer itself could or should have accomplished. I also confess to having deleted version (1)'s syntactically irrelevant leading apostrophe, which seemed to me a dead giveaway. But given what transpired in tier #3, I probably need not have worried.

word is replaced; however, tier #3 respondents were aware that the instructions to tier #2 subjects do not compel substitution in every instance.

3.3 Third Tier

I would prefer to offer herein a simple *résumé* of the results of this tier, and to reserve philosophical discussion for the concluding section. But an unavoidable methodological question now arises, that cannot be divorced from certain presuppositions; namely, with respect to which hypothesis are the quantitative data to be interpreted? And a conceptual question also arises: how—if at all—does the experiment relate to Turing's imitation game?

At least one philosopher who responded in tier #3 raised the obvious objection to the experiment (and described it as obvious even while raising it, and charitably supposed that I had already conceived a reply to it): is this experiment a Turing test at all? The question's tenor is rhetorical, for in at least one respect the answer appears clearly negative. Turing's original imitation test entailed a phase during which both human and computer were subjected to blind interrogation. Thus, to fulfil this condition strictly, our interrogators (tier #3 respondents) should have been able to ask questions of, and receive answers from, all the agents who produced the tier #2 versions of the poem. Under Turing's ideal conditions, this might have given rise to exchanges such as:

> Interrogator [to agent (2)]: "What does 'brillig' mean to you?"
>
> Agent (2): "It doesn't mean a thing to me."
>
> Interrogator: "Why did you substitute 'brilliant' for 'brillig'?"

Agent (2): "Because I was following the experimenter's instructions, and 'brilliant' is a dictionary word that resembles 'brillig'."

Interrogator: "What do you understand by the assertion 'X resembles Y'?"

Agent (2): "I understand by it that 'X appears similar to Y' in some basic way."

Interrogator [to agent (1)]: "What does 'brillig' mean to you?"

Agent (1): "It doesn't mean a thing to me."

Interrogator: "Why didn't you substitute 'brilliant' for 'brillig'?"

Agent (1): "Because I was following the experimenter's instructions, and although 'brilliant' is a dictionary word that is identical with 'brillig' in its first six characters, it does not sufficiently resemble 'brillig' to warrant the substitution."

Interrogator: "What do you understand by the assertion 'X resembles Y'?"

Agent (1): "I understand by it that 'X appears similar to Y' in some basic way."

> From these hypothetical dialogues (or dialogues which resemble them), I assert that the ideal interrogator would not be able to infer that agent (2) is human, and agent (1) is a computer. The ideal interrogator would be able to infer firstly that the agents employ different criteria in their respective assessments of the truth-value of the proposition 'X resembles Y', where Y is a dictionary word and X is not, and secondly that both agents behave in ways consistent with the experimenter's instructions and their respective assessments. Both agents are able to furnish the interrogator with plausible reasons for their respective decisions. The hypothetical computer would therefore pass Turing's imitation test.

But note that the interrogator does not require these hypothetical dialogues to draw the previous inference. In fact, many (and perhaps most) dialogues of this kind are already implied by the instruction set in tier #2, which was shown to all "interrogators" in tier #3. While instruction (5) grants permission to make a substitution, instruction (6) declares in effect that a substitution should be made only on the grounds of sufficient resemblance. Nowhere is "sufficient resemblance" defined for the interrogator, yet the notion is implicitly constrained by the information that the spell-checker functions exclusively on a word-by-word basis, which in turn implies that it ignores syntax and semantics. This information is surely inferable from the instruction set which, recall, is said to be consistent with the function of the spell-checker.

In consequence, the interrogator should conclude that both the substitution "brilliant" for "brillig", and no substitution at all for "brillig", are consistent with plausible computers, and therefore that the computer's version must be distinguished by some other means.

It follows that this experiment cannot be said *not* to be a Turing test on the grounds that it fails to allow for dialogues between the interrogator and the agents, as concerns their output. In fact, many such dialogues can be conducted implicitly by an interrogator, in the manner of thought-experiments. I claim that, by asking rhetorically whether particular substitutions are consistent with the instruction set, an interrogator can indeed eliminate versions (2) through (8).

Consider, for example:

Interrogator [thought - experimentally, to agent (2)]: "Why did you render 'And the mother rather outgoing'?"

Interrogator [thought - experimentally, for agent (2)]: "Because it is grammatical, and moreover it has meaning."

Interrogator [thought - experimentally, to agent (2)]: "I conclude that you are not the computer."

Similarly, the human authorship of version (3) is betrayed by the substitution of "gave out" for "outgrabe"; for while a computer's spell-checker might have rendered "out gave" (more likely "out

grave"), it would not have inverted the word-order for syntactic purposes. In version (4), the grammatical giveaway is "out grabbing". As well, "That was" is an incorrect expansion of " 'Twas", while "borrow" is somewhat anomalous. In version (5), the substitution "whirl" for "gyre" is synonymous—the product of a thesaurus, not a spellchecker. In version (6), the substitution of "This" for " 'Twas" is incorrect, and "thrilling" for "brillig" is quite suspect. In version (7), "mimicking" is a long way from "mimsy", but the clincher is "And the mole rashes out broke". While "mole rashes" is undeniably ingenious, the spell-checker couldn't have known that rashes "break out"—and in the past tense, "broke out". Hence, the computer could scarcely have rendered "out broke", because that is cognate with the inversion of a syntactic form which is itself embedded in a semantic context.

This leaves versions (1) and (8) as the sole possibilities. (Many tier #3 respondents arrived at this conclusion, and volunteered arguments consonant with the foregoing). But while there are persuasive reasons for selecting either (1) or (8), there is perhaps an overriding reason for eliminating version (8). The telling structural difference between these two versions is that (1) conserves the number of given words, whereas (8) does not. The latter version splits "borogroves" into "boron groves" and "outgrabe" into "out grab". While this operation may well be the kind of thing that OCR software ought to do, the operation is not in fact entailed by the given instruction set. In order to entail it, instruction (4) would have to be modified to read "If you do not find a given word in the dictionary, then try to think of a word *or words* you know, or try to find a dictionary word *or words*, that resembles *or resemble* the given word" (modifications emphasized). Instructions (5) and (6) would require similar modification. Then the operation of word-splitting would be strictly inferable from the instruction set. But as things stand, version (8) (among others) is guilty of having derived "is" from "ought".[7] By hypothetical default, then, version (1) is the remaining choice.

[7] A common feature of all the computer-generated stanzas (with the exception of one occurrence involving handwritten input) is their conservation of word number. See the appendix.

Empirically, however, the conservation of word number, which the given instructions imply, appears as a statistical non-factor in the tier #3 decisionmaking process. One-hundred-and-six respondents chose versions which conserve word number [(1),(2),(5), or (6)], while one- hundred-and-seven chose versions which do not conserve it [(3),(4),(7), or (8)]. No respondent made explicit mention of this "conservation law" and its violation by half the versions on offer.

Many respondents voluntarily communicated other ratiocinations. For example, one found "smithy troves" more *computeresque* than "slimy stoves" because it is less grammatical. This may well be the case, but then again "smithy troves" is coincidentally more poetic: it connotes images of beaten copper and hammered gold—a blacksmith's treasure-trove. Others chose version (1) primarily because of its proper nouns; these respondents were well-acquainted with spellcheckers that routinely render capitalizations. Still others (as anticipated) seized upon "Twos" as a basis for eliminating all but versions (1) and (8), but could not choose decisively between them. Then again, some respondents' reasonings were far from consistent: many indicated their second choices as well as their first, and some who selected either (1) or (8) in the first place did not necessarily select (8) or (1), respectively, in the second. Moreover, some who selected neither (1) nor (8) in the first place selected either (1) or (8) in the second.

We return to the conceptual question: how does this experiment relate to Turing's imitation game? I claim that although the experiment is not a literal Turing test in the original sense, it is another species of that same genus. This experiment constitutes a "reverse Turing test", which inquires not how proficiently a computer can imitate a human; rather, how proficiently a human can imitate a computer. And on this view, the statistical data bear further comment.

Clearly, the empirical results corroborate the argument that (1) and (8) are the sole plausible computer-rendered versions, notwithstanding (8)'s failure to conserve word number. Statistically, versions (1) and (8) were together selected with significantly greater frequency than were (2) and (7): 46% ± 2.4% for the former pair, versus 37% ± 2.1% for the latter. Then again, on an

individual basis, none of these four most popular versions was selected with statistically greater frequency than any other; their individual selection ranges all overlap. These data certainly suggest that a human can successfully imitate a computer, at least in the estimation of other humans. But the data conflict directly with the argument *ex hypothesi*, that version (1) is uniquely identifiable as the computer's. Theoretical and logical considerations indicate that versions (2), (7) and (8) should not have been selected with greater frequency than versions (3), (4), (5) and (6), because they all bear distinct marks of human fabrication.

This leads to a further question: who or what is the supreme arbiter of proficiency in such tests? On what does the credibility of an imitation ultimately depend? Turing seems to have assumed that a good correspondence would generally obtain between theoretical and empirical evaluations of a given imitation. Turing's interrogator resembles the philosopher's imaginary "man on the Clapham omnibus" and the jurist's fanciful "reasonable man". They are all incorruptible and infallible appraisers of evidence; in other words, they cannot be deceived unless, of course, the experimenter, barrister or philosopher intends that they be deceived. I hold that such a correspondence need not obtain. An imitation which the experimenter adjudges credible may be rejected by relatively many interrogators as incredible; or—as in this experiment with respect to versions (2), (7) and (8)—an imitation which the experimenter adjudges incredible may be accepted by relatively many interrogators as credible. Hence a given experiment may inform the experimenter about the nature of credibility either less or more than it is informed by him.

In our reverse Turing test, the humans who produced versions (2), (7) and (8) proved theoretically improficient yet empirically proficient at imitating the computer. This in turn suggests that many interrogators were not very proficient at gauging the credibility of the imitation. While it may be objected that the tier #2 subjects did not know the real purpose of their endeavour, this objection can be finessed by considering that in the original Turing test, the computer need not "know" that it is imitating a human. It follows that in a reverse Turing test, a human need not know that he or she is imitating a computer. Moreover, that the humans were indeed imitating a computer follows syllogistically from the premises that the humans were asked to implement a set of instructions, and that those instructions (if followed) simulated the function of a computer. Neither the computer nor the humans possessed any broader knowledge of the context itself, yet the differences in their respective functions were demonstrable.

In the concluding section, I discuss the reverse Turing test more generally, with the intention of proposing new ways—or at least new bottles for old ways—in which to illustrate differences between humans and computing machines.

4 The Reverse Turing Test

Turing posited his imitation test in a generation when computer science was nascent and computing technology comparatively primitive. In Turing's day, if one conceived of pitting a computer against a human in contests designed to measure "intelligence" generically construed, the computer was the absolute underdog. In fact, the computer was a pre-underdog, in that the technology was not advanced enough to permit such contests to take place.

Turing predicted that computers would become sophisticated enough so that, in some specified context, human interrogators would be unable to decide (with more than 30% accuracy) whether a computer or a human had rendered a given body of nominally conversational output. In other words, Turing envisioned computing progress only to the extent that human versus machine output would be indistinguishable to human interrogators. Turing seems to have supposed that a computer's ability to imitate a human would improve as a smooth function of its increased storage capacity. While current storage capacities are now remarkably close to those predicted by Turing, the computer's "cognitive" abilities have lagged far behind, to the extent that no true imitation test, in Turing's original strong sense, has yet proved feasible.

Turing's hypothesis has been weakly vindicated in many narrow contexts, notably with programs like Weizenbaum's "Eliza" (e.g. see Boden 1977, Johnson 1986). And, for example, a test group of

psychiatrists could not distinguish a transcript of the output of Colby's program "Parry", which simulates the verbal responses of a paranoiac, from transcripts of dialogues with human paranoids (e.g. Boden 1977, pp.96-111 *ff*). We generously interpret this as a success of computing, rather than a failure of psychiatry. And, for example, James Sheridan's team has "taught" a computer to compose lyrical poetry within a specified structure, such that test subjects cannot distinguish its better efforts > From poetry composed within the same structure by humans (Kern 1983, Sheridan 1987). While these examples, among others, constitute successful Turing imitation tests in a very weak sense, they naturally tend to fuel rather than to resolve the debate surrounding the strong AI thesis.

Proponents of the strong thesis, or "formalists" (e.g. Minsky 1968, Hofstadter 1981) hold that human intelligence is a property wholly explicable in terms of algorithmic complexity. Given sufficiently powerful hardware and sophisticated software, formalists believe that a computer can be built which exhibits understanding, awareness of meaning, and any and all aspects of human consciousness. They hypothesize moreover that all aspects of human consciousness consist of nothing but complex algorithms executed by a "biological computer". Opponents of the strong thesis, or "holists" (e.g. Searle 1984, Penrose 1989) hold that understanding, awareness of meaning, and other aspects of human consciousness cannot be explained solely in terms of algorithmic complexity. Holists believe that even if a computer could be built which passes any conceivable Turing test, this would not necessarily demonstrate that the computer is self-aware, that it understands what it does, or that it possesses consciousness of the human quality.

My central claim ultimately bears on this debate, but it is advanced initially on quite a different tack. On one view, progress in AI now begins to satisfy Turing's expectations, because we can conduct successful imitation tests, if but in a very weak sense. On another view, progress in computing still falls short of Turing's expectations, because there remain any number of imitation tests that the computer readily fails. Then again, on a third view, computers are able to out-perform humans in many areas, and in this sense have perhaps exceeded Turing's expectations. When it comes to performing quantitative tasks in competition with humans including playing games such as checkers and backgammon, or even chess and Go the computer is no longer underdog but overdog; not yet and perhaps never to be a Nietzschean *übermensch* in evolutionary terms (e.g. Nietzsche 1982), but demonstrably an *überhund* at parlour games.

While much of the computer's outperformance of humans is confined to various forms of "high-speed idiocy"[8] (i.e. number-crunching and the like), many humans display, by contrast, various forms of "low-speed genius" (e.g. mathematical intuition and artistic creativity). I submit that a—perhaps *the*—salient difference between computer versus human performance lies not merely in *what* they can or cannot do, rather in *how* they attempt to do what they can or cannot do. In methodological terms, the computer is an entity that strictly follows instructions, while the human is a being that constitutionally disregards them. Computers do exactly and only what they have been instructed to do, whereas humans are capable of an inexactitude that includes but is not restricted to the self-prompted or unconscious misinterpretation, omission, permutation and modification of members of a given instruction set.

In the course of this experiment, I made typically human errors in carrying out my own meta-instructions. The first involved the mistranscription of "borogroves" for "borogoves". I suppose that I too succumbed to the spellf meaning—after all, any kind of "grove" is more meaningful than every kind of "gove". My second error involved misinforming the tier #3 respondents that all the human versions were rendered by non-native speakers of English. I subsequently rediscovered in the experimental log that version (7) was rendered by a native speaker of English.

A characteristically human disregard for the tier #2 instructions was displayed by several tier #3 respondents, who chose version (2) on the grounds that it is the only version which contains all and only valid words. The instructions do not necessitate that condition. A creative human disregard for both the tier #2 and the tier #3 in-

[8]This phrase was used by Gleick (1987) to describe a dismissive attitude of some mathematicians and scientists toward computers.

structions was displayed by the two respondents who concluded that none of the eight versions was rendered by a computer. One of these two respondents argued that all the versions were rendered by human test-subjects, because the original "gyre" is a valid word which every version had replaced. The other expressed a synthetic a priori suspicion that all eight versions had been contrived by the experimenter. While these disregards affect the experiment's statistics but negligibly, they affect its conclusions significantly.

I am fairly certain that all my undergraduate students are capable, say, of carrying out the following instruction: "If you wake up in the middle of the night, make yourself a sandwich." But no robot is yet capable of carrying this out, for at least two reasons. First, the antecedent of that instruction, although decidable by humans, is not sufficiently comprehensible to humans to be made intelligible to or analogous for a computer. (What is the nature of sleep, wakefulness, dreaming, somnambulence? Like Descartes, how do you know that you are not dreaming that you are awake? Pinch yourself, and see it if hurts.) But even if we simulate the antecedent by placing our robot in "sleep mode" (an idle, low-power-consumption state) at dusk, and by programming some probability with which it will "wake" before dawn, we will be defeated by the instruction's consequent until the frame problem is solved (e.g. see Pylyshyn 1987). The generic instruction "make yourself a sandwich" can be carried out by humans only because humans are able draw necessary and necessarily self-prompted inferences from a vast store of experience and background knowledge, which a robot simply lacks. Supplying a robot with a complete set of axioms, along with a complete set of rules for correct inference-making in an epistemological—as opposed to a logical—context, is as yet an unaccomplished task.[9]

What is more telling: even if we were able to solve this multi-faceted problem, we would be assured only that if the robot "woke up" during the night, it would indeed make itself a sandwich. For while the human being understands the instruction, the price of human understanding somehow entails the possibility of disregarding. The human is capable of beginning sincerely to seek the ingredients for a sandwich, of being disappointed or distracted by the findings, and of completing the task by ordering a pizza, or by obeying any other overriding caprice.

By contrast, I am very certain that, if I instruct my undergraduate students to print their names according to the format "last name, first name, middle initial(s) if any", at least one and probably more will write in script, or will invert the ordering, or will omit their middle initial(s), and so forth. But if I instruct (i.e. program) my computer to print out a class list according to that format, then—given the data—it will do so with a negligibly small chance of making a functional error. In an overwhelming majority of such trials, the computer would execute my instructions flawlessly.

This general idea suggests a way to thwart a Turing imitation scenario. Let the interrogator give the agents instructions for the performance of some task (i.e. the generation of some verbal output). The interrogator will soon discover which agent disregards them or, commensurately, makes errors—whether intended or unintended—in their execution. That agent is human. Thus the reverse Turing test can be employed to ferret out the agents' true identities.

Note that programming a computer to output wrong answers to questions does not circumvent the reverse Turing test, for the instruction set would then have to contain a member which says, in effect, "compute the correct answer and output a different answer". The interrogator would be aware of this instruction, so an unbroken string of wrong answers would again point to the computer—for the interrogator would find that the human agent will sooner or later make a mistake and, in this case, inadvertently output a correct answer. Imagine, if you will, playing "Simon says" with a host of humans and an ideal robot. If it doesn't malfunction, the robot cannot lose; and increasingly reliable technologies diminish the likelihood of such malfunction. But even the most accomplished human player will eventually err.[10]

Naturally there are trivial cases in which the

[9]Turing (1950) recognized this problem in a section called "The Argument from Informality of Behaviour", and adroitly side-stepped it.

[10]Turing (1950) also anticipated this possibility in a section entitled "Arguments From Various Disabilities", but discounted it because his generation of computers was disposed to significant functional error.

interrogator could not distinguish the agents. For example, if the instruction set said: "Flip a coin one hundred times, and output the results in random order", then only a small proportion of humans agents would mistakenly output, say, ninety-nine or onehundred-and-one results. Similarly, a small proportion of human agents would disregard the instruction about randomizing output order, and would output the results in their obtained order (or some other order), while the randomness of the raw results themselves would preclude the interrogator's verification of their random re-ordering. But this example is utterly trivial, whereas Turing's examples of imitation tests are far from trivial, even by today's computing standards. Any useful reverse Turing test would have to be non-trivial too.

At first blush, the theses "It is conceivable that a computer can imitate a human" and "It is inconceivable that a human can imitate a computer" seem logically and empirically independent, in that the demonstrable truth of the latter appears not to condition the conjectural truth or falsehood of the former. But I claim that a deeper reading of the latter provides evidence against the former; in other words, that the reverse Turing test gives rise to an argument against the strong AI thesis.

Consider the following two syllogisms, which represent (respectively but not uniquely)[11] the formalist and holist positions:

All and only intuitively computable functions are Turing computable. (Church's thesis)
Understanding and meaning are intuitively computable functions. (formalist premise)
Therefore understanding and meaning are Turing computable. (strong AI thesis)

All and only intuitively computable functions are Turing computable. (Church's thesis)
Understanding and meaning are not intuitively computable functions. (holist premise)
Therefore understanding and meaning are not Turing computable. (contra strong AI thesis)

[11]The holistic position herein affirms Church's thesis, as does Penrose (1989). One may also espouse a holistic position by denying Church's thesis.

These arguments cannot both be sound and, if Church's thesis is false, they are both unsound. But one may suppose Church's thesis to be true (e.g. see Boolos and Jeffreys 1974). One cannot prove it true; one could only prove it false, by finding a counterexample. No counterexample has yet been found. Moreover, one can suspect that Church's thesis is true, because independent arguments lead to its equivalent statement (e.g. Turing 1937, Church 1941.) The "burden of proof" plausibly shifts to a "burden of disproof", in the absence of which we can believe the thesis confirmed until disconfirmed.

And we have reasons for supposing that understanding and meaning are not intuitively computable. The reverse Turing test furnishes one such reason. Suppose that a human (H1) is given a set of instructions (S1) which, if faithfully executed, would result in the imitation of some Turing machine (T1). But suppose that the human makes meaningful mistakes in their execution. Now we ask whether we can build another Turing machine, T2, such that T2 can similarly make meaningful mistakes. If we reply "no", then the strong AI thesis fails because Church's thesis fails, for we will have found an intuitively computable function which a Turing machine (T2) cannot perform: namely, misunderstanding, a function whose successful performance fails to imitate another Turing machine (T1). If understanding is intuitively computable, as the formalists claim, then misunderstanding should be intuitively computable too.

So formalists presumably reply "yes": we can build such a Turing machine, T2, which fails to imitate T1, and therefore which passes the Turing test in question. But whereas the human H1 fails to imitate T1 by virtue of making meaningful mistakes while executing S1, T2 must be given a set of instructions other than S1. For were T2 a universal Turing machine, T2 would execute S1 faithfully, would successfully imitate T1, would therefore fail to fail to imitate T1, and would therefore fail the test. So we must give T2 some other set of instructions, S2, whose faithful execution results in the failure to imitate T1. Then T2 would pass the Turing test in question.

But in that case, T2 would necessarily fail the associated reverse Turing test. For the reverse Turing test depends on an interrogator's exami-

nation of input as well as output. An interrogator would note that input S1, which should have led to an imitation of T1's output, failed to do so; and that input S2, which should not have led to an imitation of T2's output, succeeded in not doing so. An interrogator would then conclude that S1 had been improperly executed by a human, and that S2 had been properly executed by a Turing machine. (An interrogator could fail to distinguish the agents only in the event that the interrogator improperly executed the meta-instructions governing the reverse Turing test itself, and thus unwittingly played the human role in a hypothetical second-order reverse Turing test. This actually occurred in tier #3 herein, in the cases of the two respondents who decided that no stanza was computergenerated.)

Now a formalist could object that T2 is not on a "level playing-field" with H1. In other words, a formalist could claim that the human brain is actually running simultaneous parallel background programs (the biological equivalent of multiple "memory-resident" routines), and that mistakes in executing a given instruction set (i.e. so-called "human errors") arise from problems of interference, override, timing and other difficulties latent in parallel dataprocessing. A formalist might claim that a meaningful mistake is just a complex kind of human software "bug" or wetware dysfunction, which occurs when (putative) semantic, syntactic, analytical, emotive and other instruction sets become conflated during simultaneous execution. A formalist would claim that we can in principle program memory-resident routines in a computer that would compel it to mis-process subsequent input in apparently "meaningful" but in altogether Turing computable ways; and thus that we can in principle build a Turing machine that would tool an interrogator in a reverse Turing test.

A holistic reply to this objection is straightforward, and is consistent with the justification for assuming Church's thesis to be true; namely, that the burden of disproof lies with the doubter. Continued failure to disconfirm Church's thesis lends evidential and heuristic support to its confirmation. Similarly, we cannot prove the holist premise that understanding and meaning are not intuitively computable functions. (And perhaps we cannot disprove it either, as Searle's Chinese Room implies). But continued failure to produce even a putative disconfirmation of the holist premise lends evidential and heuristic support to its confirmation. To that support I add this modest empirical result, and the bolder notion of the reverse Turing test to which that experiment gives rise. Let anyone who denies the holist premise produce not only a set of instructions that would allow a machine to pass a strong Turing test by meaningfully manipulating tokens of natural language, but also a set of meta-instructions that would allow a machine to pass a strong reverse Turning test by meaningfully misunderstanding instructions for manipulating tokens of natural language. While no computer extant can accomplish even the former task for want of explicit instructions, mind can accomplish both tasks in the absence of explicit instructions and metainstructions alike. Until such be produced, I find no reason to discredit the holistic syllogism. Turing has yet to slay the Jabberwock.

5 Acknowledgements

I wish to thank the University of British Columbia's Centre for Applied Ethics, which provided the hardware and software for tier #1, the students at UBC who participated in tier #2, the respondents who participated in tier #3, and the respondents who volunteered other computer-generated stanzas. I would also like to thank those who afforded useful discussions about and comments on this paper; in particular, Andrew Irvine, Meg Levin, Michael Levin, James Sheridan, and the referees for *Informatica*.

6 Appendix

These versions were generated by other software packages. They rather speak for themselves.

'Twas brisling and the smithy toes
Did gyre and gamble in the wade
All missy were the borogroves
And the Mme rates outgrabe.
(MS Word 5.0)

'Was broiled and the slushy moves

Did gyre and gamble in the wage
All mimes were the barographs
And the come rates utterable.
(*FrameMaker 3.0 and 4.0, PageMaker 3.0*)

'Twos brittle and the sloths doves
Did gyre and gimbal in the wake
All mamas were the brokerages
And the home wraths outcrop.
(*PageMaker 3.0, alternative*)

'Taws brillig and the smithy toes
Did gyre and gimbals in the wade
All maims were the borogroves
And the mime rates outgrabe.
(*WordPerfect 5.1*)

'Teas brillig and the sleuth tokes
Did gyre and gamble in the wage
All moms were the borogroves
And the mode rats outgrabe.
(*WordPerfect 5.1, alternative*)

Teas Willis and the sticky tours
Did gym and Gibbs in the wake
All mimes were the borrowers
And the moderate Belgrade.
(*Apple Newton*)

References

[1] Boden, M. (1978), *Artificial Intelligence and Natural Man*, Basic Books, Inc., NY.

[2] Boolos, G. and Jeffrey R. (1974), *Computability and Logic*, Cambridge University Press.

[3] Carroll, L. (1871), *Alice's Adventures in Wonderland, and Through the Looking Glass*, Three Sirens Books, NY (undated).

[4] Church, A. (1941), *The Calculi of Lambda-Conversion*, Annals of Mathematical Studies, #6, Princeton University Press.

[5] Dennett, D. (1984), 'Cognitive Wheels: The Frame Problem of AI', in *inds, Machines and Evolution*, ed. C. Hookaway, Cambridge University Press, Cambridge.

[6] Gleick, J. (1987), *Chaos*, Viking Penguin Inc., NY.

[7] Hofstadter, D. (1981), 'A Conversation with Einstein's Brain', in D. Hofstadter and D. Dennett, eds., *The Mind's I*, Basic Books Inc., NY.

[8] Johnson, G. (1986), *Machinery of the Mind*, Times Books, NY.

[9] Kern, A. (1983), 'GOTO Poetry', *Perspectives in Computing* **3**, #3, 44-52.

[10] Marinoff, L. (1995), 'On Virtual Liberty: Offense, Harm and Censorship in Cyberspace', under review by *Computer Mediated Communication*.

[11] Minsky, M. (1968), 'Matter, Mind, and Models', in *Semantic Information Processing*, ed. M. Minsky, MIT Press, Cambridge, MA.

[12] Nietzsche, F. (1982), 'Thus Spake Zarathustra', from *The Portable Nietzsche*, ed. W. Kaufmann, Viking Penguin Inc., NY.

[13] Penrose, R. (1989), *The Emperor's New Mind*, Oxford University Press, Oxford.

[14] Pylyshyn, Z., ed. (1987), *The Robot's Dilemma*, Ablex Publishing Corporation, Norwood, NJ.

[15] Searle, J. (1984), *Minds, Brains and Science*, Harvard University Press, Cambridge, MA.

[16] Sheridan, J. (1987), 'Basic Poetry', *The Computers and Philosophy Newsletter* **1**, 83-95.

[17] Turing, A. (1937), 'On Computable Numbers, with Application to the *Entscheidungsproblem*', *Proceedings of the London Mathematical Society* (series 2) **42**, 230-65; a correction **43**, 544-6.

[18] Turing, A. (1950), 'Computing Machinery and Intelligence', *Mind* **9**, 433-460.

Computation and Embodied Agency

Philip E. Agre
Department of Communication
University of California, San Diego
La Jolla, California 92093-0503
E-mail: pagre@ucsd.edu
(619) 534-6328, fax (619) 534-7315

Keywords: artificial intelligence, planning, structural coupling, critical cognitive science, history of ideas, interaction, environment

Edited by: Matjaž Gams
Received: October 26, 1994 **Revised:** September 28, 1995 **Accepted:** October 30, 1995

An emerging movement in artificial intelligence research has explored computational theories of agents' interactions with their environments. This research has made clear that many historically important ideas about computation are not well-suited to the design of agents with bodies, or to the analysis of these agents' embodied activities. This paper will review some of the difficulties and describe some of the concepts that are guiding the new research, as well as the increasing dialog between AI research and research in fields as disparate as phenomenology and physics.

1 Introduction

From its origins in a small number of well-funded laboratories, the field of artificial intelligence has been undergoing steady structural changes:

- The field's scope has grown more precise as various neighboring fields have matured. These include disciplines such as artificial life and neural network modelling that use computational methods to study animal and human activities but that do not identify themselves as part of AI.
- AI has also witnessed the development of specialized subfields such as machine learning, natural language processing, and computational logic with their own literatures, meetings, and disciplinary cultures. These subfields develop distinctive cultures, particularly with regard to the standards by which research projects are judged.
- Research communities have arisen to apply AI methods to particular domains such as manufacturing and medicine. These communities respond to their domains in a more complex and realistic way than mainstream AI has usually done, but as a consequence they often have less freedom to explore new methods that are still poorly understood.

Many projects cross the borders among these areas of research. Many of these communities, moreover, have been heavily influenced by ideas and methods from outside AI as well, giving them a hybrid character. By choosing which disciplinary communities to associate themselves with, researchers have some flexibility in deciding, for example, whether they are engaged in scientific discovery or engineering design (or perhaps both).

As the field of AI has decentralized, its growing pluralism has made room for a variety of critical interventions and interdisciplinary dialogues. It becomes possible for groups of researchers to discover common threads in their work and to explore these collectively without needing to struggle against prestructured disciplinary boundaries or to proclaim the existence of a new, permanent institution. This article describes one such initiative, which draws together research from several fields to propose alternatives to some of the basic concepts of AI. The idea that unifies in this emerging style of research is not architectural – work is included from a remarkable variety of technical

research programs. Rather, the unifying idea is conceptual and methodological:

> using principled characterizations of interactions between agents and their environments to guide explanation and design

The theme of interaction, of course, has a long history. Systems described in the AI literature have interacted with their environments (physical or social, real or simulated) for a long time, Simon (1970: 24-25) famously pointed out that simple ants can engage in complex interactions with complicated beaches, and the concepts of cybernetics had a significant influence in the original founding of the field (Edwards 1996). The point is to bring new tools to the analysis of these interactions and to make new uses of the resulting analyses. Some rough initial explication of the key words may help orient the reader to the detailed discussions below.

interactions: The focus of this research is on activities that take place in the material world. The agents may or may not be understood as having internal mental processes that play roles in shaping the activities, but the focus is on the activities themselves.

environments: The environments in which these activities take place will generally have both physical and social aspects. The research described here, though, is primarily concerned with embodied activity in simplified versions of the physical world.

agents: The research might concern people, animals, or robots. The point is certainly not to equate people to animals or robots, but simply to establish dialogue among research projects with different goals. Serious ideas about conversational interaction and its consequences for computational modelling of human beings, for example, may inspire clearer thinking about other kinds of interaction as well.

characterizations: Attempts to conceptualize interactions between agents and their environments will require theorists to draw clear distinctions between the theorist's "aerial view" of an activity and the agent's "ground view" of that same activity. Agents that are not omniscient or omnipotent will necessarily engage in activities that are not wholly scripted, and that therefore have emergent structures that can be studied and understood.

principled: Both formal/mathematical and informal/qualitative kinds of characterizations are included. The important thing is that they be grounded in the intellectual disciplines of some field of research.

guide: It is impossible to determine in advance what forms these characterizations might take, what lessons might be learned from them, or what kinds of guidance they might offer to research. Some of this guidance will take the form of knock-down arguments, formal or otherwise, and the rest will be a heuristic matter of probabilities. Both types of guidance are valuable.

explanation and design: The goals of the research might include both scientific explanation of existing agents-in-environments and engineering design of new ones. Despite their distinct goals, the activities of science and engineering have a long history of cross-fertilization in computational research; the important thing is simply to be aware ofhich is which.

Stanley J. Rosenschein and I have recently (1995) edited a special double volume of the journal *Artificial Intelligence* that explores this approach to AI research in detail, including seventeen papers that develop the approach in particular directions. The purpose of this article is to explore some of the intellectual background to this research (Section 2), to summarize some of what has been learned from it (Section 3), and to reflect on how this research may portend the emergence of a critical cognitive science, grounded in computational experiment but simultaneously guided by critical research on its own practices and their place in history (Section 4).

2 Agents in the world: Cognition and planning

Every research community, whether it knows it or not, inherits an extensive network of ideas from

its predecessors. This inheritance may be regarded as a type of historical memory, carried across generations in the language and artifacts and interactional forms of a community, and it matters whether the memories are conscious or unconscious. Unconsciously inherited ideas will continue to shape thought and research in the present day, structuring agendas and methods and interpretations while making it difficult to conceptualize alternatives. Although it is probably impossible for any research community to become encyclopedically aware of its intellectual heritage, critical research can make a community aware of patterns that have gone unnoticed and options that it did not know it had.

This view of the role of critical reflection is common sense in many fields, particularly philosophy. Yet it is still unfamiliar in most scientific and technical fields, which are accustomed to understanding themselves as wholly aware of their own ideas and methods. The tendency of scientific fields to reconstruct their history within present-day frameworks has been long remarked (Kuhn 1962); in technical fields this sort of organized forgetting is manifested in the "state of the art" which is nowise defined by its origins. While we might be suspicious of such an assumption in chemistry or ecology or antenna physics, it is particularly implausible in the case of AI, which clearly draws upon an ancient and complex tradition of Western thought about such categories as "the mind" (Agre 1995a). Concepts of mental life have been central to AI since its beginnings; its whole premise was that computations occurring inside a computer might be regarded as modeling or mimicking the thoughts occurring inside a human being's mind.

The root metaphor here is spatial: the mind/computer as a container with an inside and an outside. Perceptions might pass into the container and willed actions might pass out of it, but the unit of analysis is the process going on inside it, as opposed to the interactions with an environment that it enters into. This way of framing AI's subject matter is understandable, given that the computers of the 1950's had only the crudest capabilities for interacting with their environments. But this framing was not a fully conscious choice; it was part of the philosophical tradition from which the psychology of that day had itself descended. Behaviorists and mentalists argued about whether it was alright to posit any mechanisms in the space between stimulus and response, but they did not argue about the stimulus-response paradigm and the container metaphor it implied.

To be sure, the lines of descent which provided AI with its root metaphors were not wholly straight. Perhaps the field's most influential founder, Herbert Simon, had previously embraced a more complex view of human beings and their lives in his writings on public administration. In his first major book, *Administrative Behavior*, Simon (1957) described numerous measures through which organizations compensate for the "limited rationality" of their members. The creation of stable job descriptions and the overall division of labor, for example, compensate for finite human abilities. Deliberately designed communication channels, likewise, compensate for individuals' limited knowledge and limited capacities to absorb new information. And hierarchical organizations provide individuals with bounded goals while providing for overall coordination. Simon portrays people and their organizational environments as fitted to one another. Metaphorically, the jagged edges of individuals' capabilities are met by the complementary shape of the world around them. Individuals are treated not as self-sufficient and self-defining but as participants in a larger organizational metabolism.

Although this theory may tend to underestimate the scope of human agency, it at least begins to comprehend people as participants in a larger world. In particular, it provides a positive theory of the attributes of that larger world which interact constructively with the complex pattern of strengths and weaknesses found in individual cognition. Yet when Simon went on to collaborate with Allen Newell in the first symbolic models of human thought (e.g., Newell and Simon 1963), the only element of it that remained is the assumption that people get their goals from their hierarchical superiors – or, more to the point, from the psychologists who are running the experiment. *Administrative Behavior* was a study of decision-making in an environment that provided the conditions for satisfactory choices despite limitations of individual rationality; *Human Problem Solving* (Newell and Simon 1972) was a study of problem-

solving in an "environment" defined in terms of the formal structure of a "problem" as a "search" through an abstract "space."

AI ideas about action have generally been framed in terms of "planning," which is roughly the notion of conducting one's life by constructing and executing computer programs (Allen, Hendler, and Tate 1990). Somewhat confusingly, this term originally entered the AI lexicon from Newell and Simon's work, but with a different meaning – it was a mechanism for shortening searches through a hierarchy of search spaces representing different degrees of detail. But its more common denotation was influenced by Newell and Simon's work as well, the idea being roughly that one constructs a plan by conducting a search through the space of possible plans, looking for the one that reaches a recognized goal state.

The development of the concept of planning provides an instructive lesson in the workings of technical ideas. The most elaborate and widely influential early articulation of it was found in George Miller, Eugene Galanter, and Karl Pribram's book *Plans and the Structure of Behavior* (1960). According to Miller, Galanter, and Pribram:

> A Plan is any hierarchical process in the organism that can control the order in which a sequence of operations is to be performed (1960: 16).

This definition is remarkably vague, speaking not of a symbolic structure but of a "process." (The process is hierarchical in the sense articulated by Newell and Simon in their own concept of "planning.") In practice, though, Miller, Galanter, and Pribram constantly shifted back and forth between two concepts of a Plan. According to the first concept, a Plan is a recipe that someone might retrieve from memory and execute as a single choice; the day's repertoire of habitual routines might be understood as a library of these Plans. This concept provides an easy explanation of why behavior has a structure: this structure is caused by a Plan that has that same structure. But it provides no explanation of how complex patterns behavior of respond to the unfolding complexities of the environment. This was the purpose of second concept, usually referred to as *the* Plan, which was a large hierarchical structure in the individual's mind, assembled from bits and pieces of Plans. At any given time the Plan would be partially assembled, well worked out in some areas and sketchy in others. The only requirement was that any given section of the Plan be completely filled in when the time comes to execute it. These two distinct concepts responded to distinct needs that Miller, Galanter, and Pribram did not know how to reconcile. Their text betrays no evidence that they were aware of the internal tension in their ideas; nor was the problem discussed in the extensive literature that they inspired. Instead, later computational research focused heavily on the construction of single Plans, with little attention to the more improvisational aspects of human activity that required the incremental assembly of the longer-term Plan.

In retrospect, much conceptual trouble in AI has arisen through a subtle tendency to conflate two logically distinct points of view: (1) that of the observer or theorist investigating an agent-environment system; and (2) that of the agent being studied or designed. Thus, for example, Miller, Galanter, and Pribam failed to distinguish consistently between two of their central theses: (1) that behavior has a structure; and (2) that behavior has the structure it does because it is caused by a Plan that has that same structure. Perhaps in consequence of this, it has become common in AI to use the term "plan" to refer either to a behavioral phenomenon or a mental entity. This usage makes it difficult to conceptualize any other explanation for the recurring structures of behavior, for example that they might arise through the repeated interaction of particular agents (which may or may not employ plans) with particular environments (which are probably arranged to facilitate beneficial forms of interaction).

The AI literature has also failed to distinguish consistently between the observer's view of the world and the agent's own model of the world. Agents are often said to possess "world models" which stand in systematic point-by-point correspondence with the outside world, and programs often receive correct, complete, consistent, up-to-date models automatically. It is of course possible that some real agents employ world models of this type, or that artificial agents might benefit from them. But the case for world models becomes less automatic once we recognize that real, situated, finite agents can only maintain models of the

world by piecing together bits and pieces of information perceived at distant places and times, often without precise knowledge of their location. Likewise, it is important to distinguish two uses of the word "situation," which can be used to refer to the totality of the state of the world at a given moment or else to the agent's own knowledge or immediate sense perceptions of the world at that moment.

Throughout this history, the unrecognized root metaphor of mind-as-container frustrated clear thinking about computational theories of action. It is clear in retrospect that action should be a promising site for reexamination of basic AI ideas, precisely because action constantly crosses the boundaries between the mind-inside and the world-outside. Yet this realization came slowly to the AI community, largely through a series of experiments with "situated" or "reactive" systems that effectively reinvented the second, neglected half of Miller, Galanter, and Pribram's ambiguous concept (Agre and Chapman 1987, Brooks 1986, Georgeff and Lansky 1987, Rosenschein and Kaelbling 1986, Schoppers 1987).

3 Structural coupling

In the context of this history, the ambition of the approach to AI that I sketched at the outset – characterizing interactions in principled ways – is to reconcile the two demands that pulled the "planning" theory in contradictory directions: explaining the sense in which activity has a structure and explaining how activity responds to a steady stream of environmental contingencies. These explanations will not be simple, nor would it be desireable to force them into a single vocabulary. Early projects in this area have been primarily concerned with mapping the territory and identifying specific, relatively modest results that can provide models for further research. This section summarizes some of these early projects, with special reference to the work described in Agre and Rosenschein (1995).

A unifying theme for this results is Maturana and Varela's (1987) notion of structural coupling. Structural coupling is a difficult concept to understand within the theories of causality that have been implicit in the majority of AI research. But it is easier to understand in its original context of evolutionary biology. Any given ecosystem, consisting of a number of species and a certain physical environment, will exhibit a great deal of mutual adaptation as the various species have coevolved while constantly having effects on their surroundings. Over time, the "design" of each species becomes increasingly interlocked with the rest of the ecosystem, so that its structure becomes well-adapted to particular forms of interaction while contributing certain ongoing influences in turn. In this sense, the structures of the various species and their environments become "coupled" – implying one another through their mutual adaptation and their roles in creating the conditions of continued existence for one another. As a result, it makes little sense to study an organism in isolation from its environment. Simple descriptive anatomy might provide a useful source of reference material, but it will not provide concepts to explain how the organism functions in its natural surroundings or why it is structured as it is.

The notion of structural coupling might be extended to other spheres, for example in understanding the systems of cultural practices by which people conduct their lives. It would be a serious mistake to reduce these practices to a simple matter of biological adaptation or survival, thereby converting AI into a type of sociobiology, but the biological metaphor has heuristic value nonetheless. Computational research on human interactions with the physical world suggests exploring the specifically cognitive dimension of these practices – the ways that they bridge the gap between the structure of an underlying architecture and the structure of an environment of activity.

3.1 Horswill

Horswill (1995) offers a framework for thinking about the adaptation of a robot's perceptual architecture to its environment. Research in AI has most often aimed at building extremely general architectures to match the seemingly infinite flexibility of human intelligence. This emphasis on generality discourages attention to environmental adaptation. Horswill, by contrast, seeks general methods for building specialized systems. Intuitively, one might imagine a lattice structure in which general-purpose systems are located toward the top and very specialized systems are located toward the bottom. The partial order that in-

duces the lattice is "works correctly in a broader range of environments than." The discovery of new forms of structure in the environment – for example, a geometric structure or property of reflected light (cf Marr 1982) – should allow the designer to move downward in the lattice, selecting a design that employs simpler machinery.

Horswill uses this framework to describe the workings of a robot that provides visitors with tours of a floor of an office building. This environment has special properties whose computational significance may not be evident at first. For example, the floors in this environment have no low-frequency surface markings since they consist of square floor tiles of uniform color, whereas all other objects do have such small markings. As a result, a simple bandpass filter, together with information from stereo matching about an object's distance from the camera, suffices to distinguish between floors and other objects. Other such environmental constraints remove the necessity of calibrating the camera, since analysis of the necessary computations reveals that, in order to make the specific decisions that it needs to make in carrying out its purpose, the robot need only calculate a function that is monotonically related to a correct measurement of the world. The final result of these discoveries is that the robot can be built with simple hardware. The point, however, is not to promote simple hardware as such, nor to suggest any particular type of hardware is universally applicable, but to provide principled means for choosing the simplest hardware that is compatible with a given environment and (desired or observed) pattern of interaction.

This method does not provide a mechanical formula for system design, since it takes considerable thought and post-hoc rationalization to discover which environmental constraints actually permit a given system's calculations to be simplified. Nonetheless, experience with this process ought to provide designers with a library of instructive case studies in both the design of adapted systems and the explanation of natural systems.

3.2 Kirsh

Whereas Horswill's design methods produce essentially passive perceptual systems, Kirsh (1995) describes and categorizes a wide variety of measures through which people actively manage their physical environments to assist their perception and cognition. Gathering the tools and ingredients needed to cook a particular dish into a specific area of the kitchen, for example, reduces the amount of visual discrimination needed to select the right object to pick up; it also provides reminders of steps that might otherwise have been forgotten. When disassembling a bicycle, arranging the parts in the order in which they were removed effectively serves as a mnemonic device to guide the process of putting them back on again (Chapman and Agre 1986). By analogy with manufacturing automation, in which workspaces are frequently provided with "jigs" that hold parts in place while other operations are performed on them, Kirsh refers to these tricks as "informational jigging."

The phenomenon of informational jigging provides numerous clues about the strengths and limitations of human cognition. It is easier, for example, to discern the length of a row than the volume of a pile. As a result, when arranging various foods on a tray it is helpful to place them in parallel lines across the countertop to ensure that one is using them in steady proportions. Capture errors are common when two common tasks provide similar patterns of visual cues; it helps to differentiate these tasks by performing them in different places or with different tools.

3.3 Hammond, Converse, and Grass

This pattern of reasoning generalizes the pattern found in Horswill's work. In each case, recourse to very general architectures is avoided by looking for structure in the relationship between agent and environment. For Horswill, this structure is a matter of perceptual patterns that permit computations to be simplified. For Kirsh, it is a matter of active interventions in the environment that produce the same effect. Hammond, Converse, and Grass (1995) take this line of reasoning further, investigating much longer-term relationships between agents and their environments. This is a striking departure from AI planning research, which has generally understood activity in terms of the pursuit of single, discrete goals.

The starting-point for Hammond, Converse, and Grass's argument is a particular kind of architecture: "case-based" systems that work by treating previously encountered situations as prece-

dents for reasoning about new ones. Case-based systems offer an explanation of why ordinary activities can proceed so smoothly despite their great complexity – most of the complexity has been encountered elsewhere before. A great deal of research has explored the memory structures needed to support case-based reasoning (Schank 1982), and Hammond, Converse, and Grass wished to apply these results to the modeling of long-term activities. Doing so, though, required some account of why newly situations are likely to be similar to old ones. The answer, in large part, is that people actively "stabilize" their environments. A simple example of stabilization is putting tools away and cleaning up when a task is finished. That way, the work environment will look largely the same at the beginning of each task.

3.4 Agre and Horswill

With the work of both Kirsh and Hammond, Converse, and Grass, the "principled characterization" of the environment takes the form of a heuristic argument and the classification of a broad range of example phenomena. Although this kind of theory is valuable, it does not support strong forms of proof. One cannot use these concepts, for example, to demonstrate formally that a particular kind of agent will necessarily enjoy all of the reminders and perceptual distinctions necessary to perform a given task. The theorist faces a trade-off: formal proofs usually require that the agent and world be understood in relatively simple and unrealistic ways.

The work of Agre and Horswill (see Agre 1995b, Agre and Horswill 1992) provides a case study in the formalization of cultural practices. They explore tasks that involve operations on artifacts, for example the artifacts found in Western kitchens during the preparation of simple customary meals. One might try formalizing these tasks in terms of the various states that each type of object might occupy (a fork might be clean or dirty; eggs might be intact, broken, beaten, or cooked; and so on) and the operations that cause objects to move from one state to another (beating an broken egg with a fork causes the egg to become beaten and the fork to become dirty). Each type of object would thus have a state-graph similar to the graphs found in conventional formalizations of planning as a matter of search. It is possible to conceive of kitchens, on another planet perhaps, in which these state-graphs are extremely tangled, so that the cook must consider a vast range of combinatorial possibilities before making any moves. But the kitchens that have evolved in human cultures do not cause such problems. An investigation of the state-graphs for familiar tools and materials suggests one reason why: these graphs fall into a small number of simple formal categories which permit a simple mechanism to determine what actions to take next.

Cooking is not always this simple, of course, but a formal analysis predicts when difficulties are likely to arise – for example when the interleaving of several complex recipes makes it necessary to schedule the use of a limited resource such as the oven. For most purposes, though, culture has evolved tools that serve cognitive functions: they help make it simple to decide what to do next, thereby reducing the need for the complex forms of state-space search that planning research has developed.

Considered together, these research projects begin to paint a picture quite different from that found in the cognitive tradition AI research. Instead of disembodied thinking, this research discovers embodied agents engaged in intricate interactions with their environments, and the properties of these interactions turn out to have substantial consequences for the agents' architectures. Just as Simon's *Administrative Behavior* had imagined organization members as fitted to their cognitive environments in complex ways that compensated for their limited rationality, this research paints human beings as fitted to their physical environments. More importantly, this research breaks down the conventional concept of cognition: cognition, it turns out, is not usefully understood as something that happens inside an individual's head. The natural unit of analysis, rather, is to be found in the interactional patterns that arrange the world as someplace that is good to think in. Research can reconstruct the structural coupling between architectures and environments by moving back and forth analytically between them, investigating the environmental structures and customary practices that might compensate for the limitations that an architecture might seem to possess when considered in isolation.

4 Critical cognitive science

The research I have described suggests an alternative conceptualization of the field of artificial intelligence. On a substantive level it suggests new units of analysis for AI, starting with the principled characterization of interactions between agents and their environments and proceeding to an exploration of the structural coupling between them. On a critical level it suggests a more sophisticated awareness of the inner conceptual workings of the field: the inheritance the field has received from its forebears and the technical difficulties that persist when this inheritance is not reexamined in the light of experience. It is conceivable that the research reported here has stumbled upon the best possible set of substantive concepts to guide future research. But this is unlikely, and it would be unfortunate to simply create a rigid new framework to replace the old one. The focus on interactions arose through reflexive study of the ideas and recurring tensions in AI research, and this habit of reflection should be codified and taught.

Although this critical approach might be brought to the design of robots or the study of insects, my own principal concern is with the study of human beings. The purpose of a "critical cognitive science," I would propose, is to employ computational techniques to study people and their lives while simultaneously cultivating an awareness of the implicit theory of humanity that this research presupposes and discovers. A critical cognitive science would be marked by the following six activities:

1. taking computational ideas seriously *as* ideas and investigating their place in the history of both ideas and institutions;

2. studying the discursive forms (the metaphors, narratives, grammar, and so on) of computational research against this same background;

3. using engineering methods as tools but doing so critically, not permitting them to import whole philosophical worldviews into the research;

4. embracing technical difficulties and impasses as potentially instructive symptoms of internal tensions in the underlying ideas;

5. critically interrogating the concepts of human beings and their lives that are implicit in technical ideas; and

6. establishing dialogue with a wide variety of other fields, according equal value to technical and non-technical interlocutors.

Measured against these standards, the research reported here provides simple, inevitably flawed starting points. The focus on interactions between agents and their environments, for example, permits this research to enter into dialogue with a wide variety of other research programs which also regard interaction as constitutive of humanity. Most of these research programs have considerably more sophisticated ideas about interaction than computational research can accommodate using the technical methods that have arisen to date. On the other hand, computational research provides powerful tools for submitting theoretical concepts to practical tests. The process of building something regularly reveals issues that have been glossed over in descriptive research, or even in formal laboratory research that has not fully enumerated the assumptions that allow laboratory phenomena to be treated as evidence for or against a theory. Critical cognitive science will have matured when the interdisciplinary dialogue routinely provides intellectual challenges in both directions.

References

[1] Philip E. Agre and David Chapman, Pengi: An implementation of a theory of activity, Proceedings of the Sixth National Conference on Artificial Intelligence, Seattle, 1987, pages 196-201.

[2] Philip E. Agre and Ian Horswill, Cultural support for improvisation, Proceedings of the Tenth National Conference on Artificial Intelligence, Los Altos, CA: Morgan Kaufmann, 1992.

[3] Philip E. Agre and Stanley J. Rosenschein, eds, Special Double Volume on Computational Theories of Interaction and Agency, *Artificial Intelligence* 72-73, 1995.

[4] Philip E. Agre, The soul gained and lost: Artificial intelligence as a philosophical project, Stanford Humanities Review 4(2), 1995a, pages 1-19.

[5] Philip E. Agre, Computational research on interaction and agency, Artificial Intelligence 72(1-2), 1995b, pages 1-52.

[6] James Allen, James Hendler, Austin Tate, eds, Readings in Planning, San Mateo, CA: Morgan Kaufmann, 1990.

[7] Rodney A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* 2(1), 1986, pages 14-23.

[8] David Chapman and Philip E. Agre, Abstract reasoning as emergent from concrete activity, in Michael P. Georgeff and Amy L. Lansky, eds, Reasoning about Actions and Plans: Proceedings of the 1986 Workshop, Morgan-Kaufmann Publishers, Los Altos, CA, 1986.

[9] Paul Edwards, The Closed World: Computers and the Politics of Discourse, MIT Press, 1996.

[10] Michael P. Georgeff and Amy L. Lansky, Reactive reasoning and planning, Proceedings of the Sixth National Conference on Artificial Intelligence, Seattle, 1987, pages 677-682.

[11] Kristian J. Hammond, Timothy M. Converse, and Joshua W. Grass, The stabilization of environments, Artificial Intelligence 72(1-2), 1995, pages 305-327.

[12] Ian Horswill, Analysis of adaptation and environment, Artificial Intelligence 73(1-2), 1995, pages 1-30.

[13] David Kirsh, The intelligent use of space, Artificial Intelligence 73(1-2), 1995, pages 31-68.

[14] Thomas S. Kuhn, The Structure of Scientific Revolutions, Chicago: University of Chicago Press, 1962.

[15] David Marr, Vision, San Francisco: Freeman, 1982.

[16] Humberto R. Maturana and Francisco J. Varela, The Tree of Knowledge: The Biological Roots of Human Understanding, Boston: New Science Library, 1987.

[17] George A. Miller, Eugene Galanter, and Karl H. Pribram, Plans and the Structure of Behavior, Henry Holt and Company, 1960.

[18] Allen Newell and Herbert A. Simon, GPS: A program that simulates human thought, in Edward A. Feigenbaum and Julian Feldman, eds, Computers and Thought, McGraw-Hill, 1963, pages 279-296.

[19] Allen Newell and Herbert A. Simon, Human Problem Solving, Englewood Cliffs, N.J., Prentice-Hall, 1972.

[20] Stanley J. Rosenschein and Leslie Pack Kaelbling, The synthesis of digital machines with provable epistemic properties, in Joseph Halpern, ed, Proceedings of the Conference on Theoretical Aspects of Reasoning About Knowledge, Monterey, CA, 1986.

[21] Roger C. Schank, Dynamic Memory: A Theory of Reminding and Learning in Computers and People, Cambridge University Press, 82.

[22] Marcel Schoppers, Universal plans for reactive robots in unpredictable environments, Proceedings of the Tenth International Joint Conference on Artificial Intelligence, Milan, 1987, pages 1039-1046.

[23] Herbert A. Simon, Administrative Behavior: A Study of Decision-Making Processes in Administrative Organization, second edition, New York, Macmillan, 1957.

[24] Herbert A. Simon, The Sciences of the Artificial, Cambridge: MIT Press, 1970.

Methodological Considerations on Modeling Cognition and Designing Human-Computer Interfaces — an Investigation from the Perspective of Philosophy of Science and Epistemology

Markus F. Peschl
Dept. for Philosophy of Science
University of Vienna
Sensengasse 8/10, A–1090 Wien
Austria, Europe
Tel. +43 1 402-7601/41, Fax: +43 1 408-8838
E-mail: a6111daa@vm.univie.ac.at.

Keywords: cognition, epistemology, HCI, knowledge representation

Edited by: Matjaž Gams
Received: May 15, 1995 **Revised:** October 30, 1995 **Accepted:** December 4, 1995

This paper investigates the role of representation *in both cognitive modeling and the development of human-computer interfaces/interaction (HCI). It turns out that these two domains are closely connected over the problem of knowledge representation. The main points of this paper can be summarized as follows:*
(i) Humans and computers have to be considered as two representational systems which are interacting with each other via the externalization of representations. (ii) There are different levels and forms of representation involved in the process of HCI as well as in the processing mechanisms of the respective system. (iii) As an implication there arises the problem of a mismatch *between these representational forms – in some cases this mismatch leads to failures in the effectiveness of HCIs.*
The main argument is that representations (e.g., symbols) typically ascribed to humans are built/projected into computers – the problem is, however, that these representations are merely external manifestations of internal neural representations *whose nature is still under investigation and whose structure seems to be different from the traditional (i.e., referential) understanding of representation. This seems to be a serious methodological problem.*
This paper suggests a way out of this problem: first of all, it is important to understand the dynamics of internal neural representations *in a deeper way and seriously consider this knowledge in the development of HCIs. Secondly, the task of HCI-design should be to* trigger appropriate representations, processes, *and/or* state transition *in the participating systems. This enables an effective and closed feedback loop between these systems. The goal of this paper is not to give detailed instructions, "how to build a better cognitive model and/or HCI", but to investigate the epistemological and representational issues arising in these domains. Furthermore, some suggestions are made how to avoid methodological and epistemological "traps" in these fields.*

1 Introduction – setting the stage

Designing a model of cognition or a human-computer interface is normally considered to fall into the domain of computer science. Techniques and strategies from computer graphics, software engineering, etc. are assumed to be the foundation and the starting point for developing a human-computer interface. Contrary to this view, the goal of this paper is to show that issues from

cognitive science (e.g., [62, 55, 71, 73] and many others), *epistemology* (e.g., [45, 4, 7, 11, 38] and many others), as well as from *philosophy of science* (e.g., [8, 27, 16, 5] and many others) are at least as important as the technical questions which are covered by computer science. An even more radical approach is suggested: before one can even start to think about a human-computer interface, he/she has to consider and investigate a much more fundamental level, which – only at first glance – seems to be completely detached from the original idea of implementing a human-computer interface. This fundamental level concerns the epistemological question of *knowledge (representation)*. In the course of this paper a perspective will be developed in which knowledge representation is the implicit theme on which all activities in human-computer interface development are based. The goal of this paper is to make explicit all the different levels and forms of knowledge representation which are involved and interacting with each other when a cognitive system interacts with a computer. The two main claims of this paper can be summarized as follows: in order to develop a successful and adequate human-computer interface two criteria have to be fulfilled:

(i) one has to be clear about the *epistemological situation* in which he/she finds him-/herself when either developing or using a computer (interface). In other words, there has to be some clarity about the different forms, levels, and dynamics of knowledge which are interacting with each other when someone uses a computer.

(ii) only if one has an *adequate cognitive model*, it is possible to create an effective and efficient interaction between human and computers. I.e., only a sound theory about the dynamics of the cognitive processes being involved in human-computer interaction guarantees a successful interface between these two systems.

1.1 Humans, computers, nervous systems, and knowledge

Before tackling the questions which are arising in the context of human-computer interfaces, I am suggesting to take a step back and look at the whole problem from a more abstract, fundamental, and epistemological perspective. In most approaches this step is either regarded as uninteresting or it is ignored at all. However, in order to achieve a clear view of the problems being involved in designing a model of cognition and using a computer interface the following considerations will prove very useful in the sections to come.

So, what is the basic situation in which we find ourselves when developing a human computer interface and/or a model of cognition? First of all, we have to be clear about the participating systems which are involved in this interaction: (i) the *user* who can be characterized as a *cognitive system* which tries to solve some problem or to accomplish a task more efficiently by making use of a computer; (ii) the *computer* which can be characterized as a machine *transforming* inputs into outputs in a non-linear manner[1]; (iii) the *interaction media* in the computer allow the interaction between the cognitive system and the computer. There exists a wide range of input/output devices: mouse, keyboard, printers, graphic displays, acoustic in/output devices, data glove, etc.; (iv) one must not forget that the user has also his/her interaction devices, namely his/her *sensory* and *motor systems*. They allow external stimuli (such as pixels on a screen) to enter into the (neural) representation system and that internal representations can be externalized via behavioral actions. These behavioral actions (might) change the environmental structure (e.g., move the mouse, hit a key of the keyboard, etc.); (v) finally there is an *observer* – in most cases this observer is also the designer of the human-computer interface and/or of the cognitive model. He/she has access to both the internal structures of the computer program and to the behavioral structures of the user. We have to keep in mind that the access to the user's internal representational structures is only *limited*, as the user can only externalize a small fraction of his/her knowledge via behavioral actions, such as language. It is the task of the designer to develop an adequate model of the cognitive processes and of the potential user's internal representations (and their dynamics) by making use of neuroscientific theories and findings from cognitive science.

Investigating the processes which are going on

[1]Of course, cognitive systems can be characterized as *transformation systems* as well – and this paper assumes that this is the case.

in human-computer interaction, it is clear that we are not dealing with a one-way interaction, but with systems which try to *mutually influence* and *trigger* each other in a more or less beneficial manner. As it is the case in almost any interaction between a cognitive system with its environment or with other cognitive systems, we are dealing with a *feedback relationship* – the goal of this relationship is to establish a more or less stable feedback loop being based on a "smooth" interaction and on effectively triggering the respective representation/processing systems.

By now its has become clear that a lot of interactions are going on between these two systems (i.e., the human and the computer). Consequently, there have to be devices which act as *interfaces* between these two systems which – at first glance – do not seem to be compatible. In other words, how is the interaction between the user's and the computer's dynamics realized? The answer to this question covers a large part of what the field of human-computer interaction is all about. Let's have a short look at the interaction media which are involved in this process of interaction: (a) the user's motor system (e.g., hand, voice, etc.), (b) the user's sensory system (e.g., visual system, tactile receptors, acoustic system, etc.), (c) the computer's input devices (e.g., keyboard, mouse, data glove, etc.), and (d) the computer's output devices (e.g., graphical displays, all kinds of virtual reality output devices, etc.). These interaction media are responsible for creating some kind of *compatibility* between the internal representations of the participating systems. Their task is to transform the internal representations into structural changes of the environment (e.g., activating a muscle which moves a mouse, activating a pixel on a screen, etc.) and vice versa. The human and the computer can only interact with each other via mutually changing the environmental structure/dynamics (e.g., generating sound, pressing a key on the keyboard, etc.). Similarly as in (natural, spoken, or written) language, communication is only possible by making use of the environment as *carrier* for the mutual stimulations.

1.2 Properties of the participating parties

In order to understand the processes occurring in the interaction between humans and computers, we have to be clear about the "epistemological context" in which these interactions take place. Therefore, before designing cognitive models and/or HCIs the participating systems and their (representational) dynamics have to be investigated more closely (see the following subsections). The focus of our attention should be the *(human) cognitive system* and its *representational function*, as it is the "main player" setting the boundary conditions in these interaction processes. Secondly, the *environment* ("world") has to be considered: every cognitive system is embedded into and has to survive in this world by making use of its representations of the environment. Thus, we have to study the representational relationship between the cognitive system and the environment (see sections 1.3 and 2)[2]. It turns out that *language* and symbol systems (in the broadest sense) play an important role in the problem of knowledge representation. In the course of the section to come it will become clear. however, that these linguistic/symbolic representations are *embedded* in a more general and more flexible representational substratum, namely neural systems. In a last step the epistemological properties and role of *computers* (as representation systems and simulation machines) have to be studied (see section 1.2.4). Only then we will be able to understand the processes and problems which are arising in the context of modeling cognition and developing HCIs.

1.2.1 Cognitive system/user

The central part of human-computer interaction is the *cognitive system* which is not only interacting with the computer, but also (and for the most part) with the remaining environment as well as with other cognitive systems. From observing a cognitive system which is acting (successfully) in its environment one can conclude that this system must possess some kind of *knowledge* about its environment. Otherwise it would not be possible to

[2]This is an essential epistemological *requirement* for developing an adequate model of cognition and, consequently, an effective HCI.

behave adequately in the environment[3]. Cognitive science as well as (cognitive) psychology assume that cognitive systems *represent knowledge* about the environment and about how to successfully interact with this environment. More specifically, a representation system is postulated to hold some kind of information about the environment and how to *survive* in a given internal and external environmental context. Furthermore, these "cognitive disciplines" assume that the cognitive system makes use of its *representation system* and the representations in order to generate adequate behavior (e.g., [3, 7, 62, 55] and many others).

"Adequate behavior" and "survival" are used in a very wide sense: to externalize adequate behavior refers to behavior which ensures survival in a physical (e.g., finding food), social, linguistic, cultural, or even scientific context. The goal of a cognitive system can be characterized as the attempt to establish a *stable (feedback) relationship* both inside the organism and with the environment (compare also to the concepts of *homeostasis* and *autopoiesis*, e.g., [48, 73] and many others). In humans and most other natural cognitive systems the *nervous system* is assumed to be the substratum for the representation system. I.e., the neural architecture (as well as the body structure [58, 59]) holds/*embodies* all of the particular human's/cognitive system's knowledge. Thus, it is responsible for his/her/its behavioral dynamics.

1.2.2 Environment

Every cognitive system is embedded into the environment. Abstractly speaking, the environment can be characterized as a complex system of flows of energy consisting of *meaningless patterns* and *regularities*. In the perspective being presented in this paper the term "environment" refers to I.Kant's concept of the "thing-in-itself". It is not accessible *in principle* and – despite of all efforts of science to find out more about the "true" or "objective" nature/structure of the environment – we can perceive only *representations/constructs* of the environment; i.e., representational constructs are generated by our nervous system in the course of interacting with the environment as well as with its neural states.

It is only in the process of interacting with a cognitive system that environmental states/patterns receive *individual meaning*. According to G.Roth meaning or semantics is the specific influence or the effect which a environmental state/dynamics has on a specific cognitive/representational system [63, 64]. Thus, meaning is always *system-relative* and *individually* depends on the structure and current state of the particular cognitive system. This structure/state itself is the result of all phylo- and ontogenetic developments of the particular organism (i.e., the total of the organism's "experiences").

Having in mind what has been said about the impossibility of accessing the environment directly, it has to be clear that the same applies to what has been referred to as "environmental regularities". I.e., environmental regularities do not present themselves explicitly as regularities; in other words, it is the organism's task to figure out *these* regularities which are *relevant* for *its survival*. This is not only true for simple organisms which are using, for instance, light gradients for locating food more efficiently, but also for scientists who are trying to find out specific regularities in the environment in order to use them for manipulating the environment more efficiently. The important thing to keep in mind is that all these regularities are *system relative/-dependent* and are the result of *construction* processes which are executed by the particular representation system[4]. Looking more closely at the structure of the environment, it turns out that one has to differentiate between two forms of regularities with which cognitive systems are confronted:

(i) "*natural regularities*": this category includes all regularities which are occurring "naturally" in the dynamics of the environment. The fact that a stone will always fall down or that lightning is followed by thunder are examples for such "natural regularities".

(ii) "*artificial regularities*": this category of regularities can be referred to as *artifacts* in the

[3]Of course, there is always the possibility to behave in a random manner. For obvious reasons this seems to be a rather bad strategy to ensure the organism's survival.

[4]This implies that even so-called "objective" or "true" *scientific knowledge/theories* are only system relative and always depend on the structure of these cognitive systems which are responsible for constructing them.

broadest sense. They can be characterized by the fact that they are the result of *externalizations* (behaviors) of an organism's knowledge. In other words, artifacts are artificial changes or alterations in the structure of the environment. The notion of artifact, as it is used in this paper, is rather wide and ranges from simple forms of tools, shelters, houses, etc. (of simple animals as well as of humans) to the most advanced technological achievements or scientific theories. Everything which has been produced by a single organism or a group of cognitive systems is included in the domain of artifacts. It is clear that artifacts follow the same dynamics and rules (of physics) as natural regularities do – the difference is that they are carrying an additional structure/regularity/feature which has been attached to them by an organism's behavioral action. Of course, it is sometimes difficult to clearly differentiate between artificial and natural regularities.

We are dealing with *two interacting dynamic systems* whenever we are studying the interaction between a cognitive system and its environment. Both systems are following their own dynamics and try to influence and modulate each other. Especially the cognitive system tries to achieve a state of homeostasis (i.e., the criterion for life and/or survival) by externalizing certain behaviors which modulate the internal and external environment in a beneficial manner. Cognitive systems are a part of the environment. A cognitive system itself can be characterized as consisting of the following subsystems which are heavily interacting with each other: (a) the body structure and state of activation patterns which are responsible for the generation of the actual behavioral dynamics; (b) the structure of the synaptic weights which are responsible for holding the organism's knowledge and which, when changed, are responsible for the phenomenon of ontogenetic "learning" and/or adaptation; (c) the genetic code and dynamics underlies all these activities. It regulates the phylogenetic development of the organism (as well as of its [phylogenetic] knowledge). In any case, a complex pattern of interactions and different levels of knowledge are involved in this interaction between cognitive systems and their environment (see [59] for further details). The basic assumption is that a cognitive system has to hold some kind of information or knowledge about its environment in order to survive in this environment.

1.2.3 Language and symbols

A subgroup of artifacts has a special function: it is used for representation, communication, and storage of *knowledge* and information. Especially so-called higher organisms are using these artifacts for transmitting knowledge to other organisms via an extra-genetic path. I.e., normally it is only possible to pass on knowledge to another generation of organisms through the genetic code. Of course, all the knowledge which has been accumulated in the course of the organism's ontogenetic development is lost in this process. In order to cut short the – sometimes painful – process of having to make "direct experiences" in the environment, a kind of *symbol system* is introduced which describes these experiences in an abstract manner [36]. If another organism is capable of decoding these messages, it can "learn" from these symbolic artifacts instead of having to directly experience the environmental consequences of its behavioral actions. The important property of these artifacts is their *referential* function; i.e., they are symbols in the most general sense, meaning that they are environmental regularities which are referring to something else. This subgroup of artificial regularities includes all kinds of language (written, spoken, sign language, body language, etc.), symbols, books, paintings, TV, CDs, scientific theories, architecture, etc. Almost any artifact can be interpreted as having some kind of referential function, namely it refers at least to what the creator of this artifact has intended to express/externalize. In symbols [17, 18] this referential function can be seen most clearly: the environmental pattern of a symbol s *stands for* another state/pattern e in the environment or in an organism. In other words, the symbol s *represents* e.

This kind of artifacts can be understood as being the *substratum* of what is referred to as "*cultural knowledge*" in the widest sense. It is the basis for any cultural process and development. Keep in mind, however, that even these artifacts are completely *meaningless* per se! The same applies for them as for any other environmental regularity or pattern: they receive their particular meaning only in the process of being interpreted by a cogni-

tive system, where its representational dynamics is modulated/influenced by this symbol. Their meaning is by no means clearly defined; rather it always depends on the structure, state, and phylo- and ontogenetic experiences of the perceiving cognitive system. In other words, their meaning/semantics is always *system relative.* For a human reader a book will have a different meaning than for an insect which is interested in eating paper. But even between humans a certain piece of text or spoken language will have different meaning – it always depends on the previous (learning) experiences and on the current (representational) state of the participating cognitive systems which meaning is attributed to an artifact. This problem of "private semantics", communication, and its consequences for AI and cognitive science will be taken up again in the sections to come.

1.2.4 Computer and its program

A special subgroup of this (referential) subgroup of artifacts contains *computers.* They are explicitly designed to *transform information* and *knowledge.* The designer's task is/was to build a mechanism (i.e., artifact) which supports humans in accomplishing a certain task at a higher speed and/or with higher accuracy by making use of and manipulating referential artifacts. Normally a human would use his/her own representational system (and body) in order to fulfill a certain task. The whole concept of computers is based on the idea to transfer parts of his/her representational structure (i.e., knowledge to solve a certain problem or task) into a computer program which – by making use of these knowledge structures – is then capable of *mimicking* certain cognitive activities at some level of abstraction[5]. The computer runs these programs automatically by doing nothing, but transforming and manipulating bit patterns according to certain rules (i.e., algorithm). As it is the case with any other artifacts and environmental regularities, it is only the act of *interpretation* by a human that brings meaning to these meaningless bit patterns (e.g., pixels on the screen are perceived as meaningful symbols, computer generated sound-waves are interpreted as spoken language, etc.). In other words, the computer's output *triggers* and *modulates* the cognitive/representational dynamics of the human user who is interacting with the computer (and vice versa).

The epistemologically interesting part of this interaction is the process of *transferring knowledge* from the cognitive system (e.g., an expert, human, etc.) to the representational structure of the computer (e.g., data structures, algorithms, rule systems, semantic networks, neural networks, etc.). More precisely, the question is, how the computer and its representational structure obtain their knowledge allowing them to solve a problem or to achieve a certain task. There are at least two answers to this question – they do not mutually exclude each other:

(i) The knowledge is *transferred* from the human (expert) to the computer. In other words, some kind of mapping between the human's representation system and the computer's representation mechanisms (e.g., data structures, programs, etc.) is carried out. That is the usual procedure on which most of the current *knowledge engineering* techniques are based. The human/expert has to make (learning-) experiences in the real world by actively interacting with the environment. By doing so he/she constructs knowledge and theories about the environment. This knowledge is externalized by using language. These linguistic expressions are formalized (e.g., by a knowledge engineer or a programmer) and transformed into an algorithm, computer program, and data structures. Hence, the computer makes use of already *prefabricated* representations.

What makes this approach interesting is that the computer (program) can handle huge amounts of data which cannot be overlooked by humans, it can manipulate data with extremely high speed, and thereby make implicit structures explicit (e.g., the solution of differential equations, the application of rules to a set of input data, etc.). From an abstract perspective an expert system using a rule based knowledge representation mechanism is pretty uninteresting. The space of possibilities/solutions is already *predetermined* by the set of rules as well as by the possible/acceptable input data. What makes these systems interesting is the fact that this

[5]This does not only apply to expert systems, but to any computer programs which perform a certain task.

space is extremely large and that it is – for humans – almost impossible to foresee all solutions. The computer's ability to stupidly follow the rules and apply them to the data with high peed makes these structures, which are *implicitly* given in a set of rules, *explicit*. This process generates results (i.e., particular states in the "knowledge space") which are (might be) interesting and/or helpful for humans. They are interesting, because the user could not have reached this solution by applying his/her knowledge. Of course, he/she could have done exactly the same as the computer (namely following a huge set of rules), but this approach would have been too time consuming and, thus, not worth pursuing.

(ii) An alternative approach is that the *computer itself* "learns" from its experiences with the environment in a *trial-&-error* process. That is the way which most approaches in the domain of *artificial neural networks, computational neuroscience* (e.g., [65, 49, 34, 14] and many others), and of *genetic algorithms* (e.g., [35, 31, 50] and many others) follow. The basic idea can be summarized as follows: in the beginning of the learning procedurethe computer has (almost) no useful knowledge (to fulfill the desired task). I.e., its behavior follows more or less random patterns. Neural learning algorithms or genetic operators *adapt* the representational structure (i.e., synaptic weights, genetic code, etc.) in a trial-&-error manner so long, until some useful/desired behavior is generated by the representational structure.

This is similar to the processes which occur, whenever a human or any natural system has to learn something. He/she/it *adapts* to certain environmental regularities which are useful for the organism's survival in order to make use of them in a beneficial manner. In any case the result is a representational structure (in the brain or in the computer) which is said to be capable of dealing successfully with certain aspects of the environmental dynamics in the context of accomplishing a certain task, such as the organism's survival or solving a problem. The difference to solution (i) is that no prefabricated chunks of knowledge are mapped/transferred to the representation system – the cognitive/computer system rather has to figure out a way of solving a certain problem by adapting its knowledge structures in a trial-&-error process.

In any case the implicit assumption is that the resulting knowledge structure has some kind of resemblance or even *iso-/homomorphic* relationship to the environmental structure. Looking more closely at this postulate, it turns out that this implies some kind of homomorphic relationship between the structure of the environment, of the representation in the (human) cognitive system, as well as of the representation in the computer. It is argued that due to this relationship of (structural) similarity it is possible that humans can solve the problem of their survival in the environment. Furthermore, if humans can solve problems with this kind of "structure preserving" representation, computers can do similar things by applying the same representational mechanism.

However, most models in traditional (i.e., symbol manipulation) cognitive science as well as in traditional AI have not been as successful as originally promised! The success of AI has been limited to rather specialized and highly formal domains. For the remaining part of this paper the reasons why AI-models have not been so successful will be discussed. Furthermore, the implications of these problems for models of cognition and human-computer interfaces will be investigated.

1.3 Epistemological questions concerning the traditional concept of representation

From an epistemological perspective two problems seem to give an explanation to why traditional cognitive models and human-computer interfaces have not been as successful as originally assumed. These problems are rooted in an inadequate assumption about knowledge representation:

(a) Epistemological as well as neuroscientific evidence gives rise to the conclusion that the postulated homomorphic or mapping representational relationship has to be questioned or even given up. I.e., it is *implausible* to make the assumption of a structural correspondence or iso-/homomorphic relationship between the structure of the environment, the human's representation of the world, and the structure of the representation in the computer.

(b) As an implication of (a) it becomes clear why we will encounter problems in the interaction between humans and computers. As there are structural differences in the participating representational mechanisms, there will be a lack of compatibility. This leads to an inefficient interaction between human and computer representations, as the participating forms of representations do *not* fit into each other and/or are mutually not compatible. Think, for instance, of a symbolic or graphical user interface: it will have only limited success, as in many cases it will not meet the requirement of adequately triggering, modulating, and influencing the dynamics of the neural representation system (and vice versa).

As a major implication of these problems it follows that we have to study the properties, structure, and dynamics of the participating (neural/human) representational systems *first.* In other words, we have to find theories about how knowledge is represented and transformed in the neural system, in order to modulate and manipulate the very same neural system in an efficient manner e.g., by letting it interact with a computer. Only then can we start developing cognitive models, so-called knowledge-based systems, and user interfaces! This is the *basic requirement* for any kind of "user-friendly" interaction with a computer. Abstractly speaking, the goal of human-computer interfaces can be defined as *triggering and modulating the user's representational system efficiently.* As we have seen in the previous section, we are confronted with two complex dynamic systems (i.e., the computer and the brain) having internal states and following their internal dynamics, which are interacting with each other. Only, if one knows the internal structure (i.e., the structure of state transitions) of both systems, it is possible to influence the state transitions of the respective system efficiently[6].

As one can change the structure and dynamics of a program rather easily and as one (normally) knows the structure of state transitions of the computer program, the program should adapt to the cognitive/representational structure of the users, rather than the other way around. The goal should be at least that the need for changes in the user's cognitive structures should be kept to a minimum. Considering the issues in the sections to come could be a first step towards achieving this goal.

[6]In fact, that is not only what developing human-computer interfaces is all about, but also any kind of communication or even advertising.

2 Troubles with traditional approaches to knowledge representation

2.1 Propositional vs. pictorial representation

Traditional cognitive science, AI, and cognitive psychology offer two main paradigms for knowledge representation: (i) *propositional representation* (being based on works by [22, 23, 24, 53, 51, 52, 75] and many others) and (ii) *pictorial/depiction representation* (i.e., mental imagery being mainly based on works by [42, 39, 40, 41, 68] and many others). In the course of AI's short history propositional representations had a much stronger influence than any kind of pictorial representation, as they are far more practical and useful for the task of representing and manipulating complex knowledge structures. However, pictorial representation plays a central role in the context of (graphical) human-computer interfaces.

There exists a long ongoing discussion between these two approaches (e.g., [42, 41]) – the goal was to show the basic differences between these two concepts of representation. The sections to come do *not* follow these discussions. Rather, the idea is to show two points: (a) both approaches are – from an epistemological as well as neuroscientific perspective – *naive* and rather *insufficient* as adequate models for cognitive processes and as representational concepts. (b) These two approaches are *not* as different as they might appear at first glance. In the following paragraphs it will turn out that both the pictorial and the propositional concept of knowledge representation are – on a more fundamental and epistemological level – based on very similar basic assumptions and premises. Especially the underlying understanding and implicit assumptions about representation (i.e., the idea of a referential representational relationship) are more or less the same. Furthermore, the shortcomings and problems arising from

these considerations in the context of cognitive modeling and of developing a human-computer interface will be discussed.

2.2 Questioning the referential concept of representation

2.2.1 Ambiguity in the process of interpretation and of transferring knowledge

From the field of knowledge engineering, of programming, and of logic it has become clear by now that in the process of extracting knowledge from an expert and transferring it into a computer a lot of information is *lost* for various reasons (e.g., certain parts of the knowledge cannot be verbalized, cannot be formalized, etc.). What seems to be more important, and what seems to be neglected in many cases, is the fact that not only is information lost in this process, but that the *semantics* is also *altered* or *distorted* in many cases. In fact, it seems that the so-called loss of information is only an extreme case of a change in the semantics. This does not only apply to symbolic knowledge representation, but also to pictorial representation (e.g., visual ambiguities, etc.). This seems to be a problem; not only for expert systems, but also, and perhaps especially, for human-computer interfaces, as most of these "semantic shifts" occur at this critical step when one form of knowledge representation is transformed into another. What are reasons for this phenomenon of semantic shifts?

(a) *natural language* is one of our main instruments to externalize our (internal) knowledge. As is shown by [60] (and by many others) and as everybody knows from his/her own experience, any kind of language is capable of externalizing only a *small fraction* of the semantics which one has in mind when he/she tries to express something by making use of his/her language. The "tacit" or "implicit" knowledge is not only lost in the moment of externalization, but also some kind of semantic distortion occurs: due to his/her different onto- and phylogenetic experiences the receiver of the externalized language will interpret these "meaningless syntactic environmental patterns" (see section 1.2.3) in a different way as the sender of the message.

(b) Thus, a semantic shift occurs, which *cannot* be avoided in principle, whenever one is externalizing (symbolic) behavior and somebody else tries to interpret these – per se – meaningless artifacts. This implies that the semantics in different users and/or designers and/or experts might differ considerably. Although they are confronted with the same symbol, icon, graphical representation, etc., these representations might trigger different internal representations/semantics in the participating brains.

(c) This distortion is taken even one step further in the process of *formalizing* natural language into purely syntactic and formal structures. Despite all attempts to introduce "semantic features" into symbol systems, natural language is deprived of its semantic features and dimension in the process of formalization (and, in general, in the process of externalization). Symbolic representations (as well as pictorial representations) remain syntactic in principle. Loosing the semantic dimension implies, however, more freedom in the process of interpreting these syntactic/formal structures which, in turn, may lead to unwanted semantic shifts.

(d) In most artificial representation systems a lack of *symbol grounding* can be found. Semantics is assumed to be somehow externally defined or given. Furthermore, it is assumed that the semantics is more or less stable over time. Epistemological considerations as well as our own experiences reveal, however, that (i) semantics changes individually in minimal increments (according to the experiences which he/she makes with the use of certain symbols). (ii) There is no such thing as "the one given semantics"; public as well as private semantics are in a steady flow. As we have seen, the semantics of symbols is always system relative and communication is based on mutually adapting the individual use of symbols (compare also the concept of a consensual domain as basis for a public semantics; [6, 28, 29, 46, 48]). Consequently, the idea of holding the semantics stable is absurd anyway – knowledge representation techniques rather should provide means which deal with the phenomenon of an "individual experience-based adaptive semantics".

(e) As has been mentioned already, a major distortion of semantics occurs in the process of transforming one form of representation into another. I.e., in the process when an internal representation is externalized and received by another system and transformed into its internal representational format. This happens in any human-computer interaction. The problem here is that – contrary to human-human interaction/communication – it is almost impossible for both parties to ask whether the respective system really "understood" what the other was trying to express. This is due to the (false) implicit assumption that our language and even our pictorial/iconic representations are based on a stable and somehow "given" semantics.

2.2.2 Mapping the environment

Both in propositional and in pictorial representation the underlying idea of representation can be characterized as follows: the environment is *mapped* more or less *passively* to the representational substratum. Although most approaches in this field distance themselves from the idea of a naive mapping (i.e., naive realism), an *unambiguous stable referential/representational relationship* between the structure of the environment and the structure of the representational space is assumed. In other words, a symbol or a (mental) image *refers to*, represents, or stands for a certain phenomenon, state, or aspect of the (internal or external) environment.

Empirical research in neuroscience gives evidence that *no* such stable and unambiguous referential relationship between repraesentans (i.e., the representing entity) and repraesentandum (i.e., the entity to be represented) can be found[7] [37, 14, 69]. It seems that neural systems do *not* follow this assumption of a referential representational relationship. As is discussed in [57] there are not only empirical, but also epistemological and system-theoretical reasons as to why the concept of referential representation does not apply to neural systems. It can be shown that in highly recurrent neural architectures (as our brain) neither patterns of activations, nor synaptic/weight configurations, nor trajectories in the activation space refer to environmental events/states in a stable (referential) manner. This is due to the influence of the *internal state*[8] on the whole dynamics (as well as on the input) of the neural system. As an implication it is necessary to *rethink* the representational relationship between the environment and the representation system. This is not only important for the development of adequate models of cognition, but also for designing human-computer interfaces, as their design is based on assumptions stemming from a referential understanding of representation (e.g., icons, symbols, images of desktops, etc.)

2.2.3 Is depicting the environment sufficient?

In the process of studying the phenomenon of representation two aspects and functions of representation have to be taken into account: (i) *mapping* or *modeling* the environment in the representational structure; i.e., the goal is an adequate and accurate model, picture, description, etc. of the environment; (ii) *generating (adequate) behavior*: an equally important task of a representation system is to generate behavior which allows the system to accomplish a certain task (e.g., its survival or solving a problem).

Both in the propositional and pictorial approach the aspect of *mapping* the environment to the representational substratum is more important than the aspect of generating behavior. The implicit assumption of these approaches runs as follows: if the environment is represented/depicted as accurately as possible, then it will be extremely easy to generate behavior which adequately fits into the environment (i.e., which fulfills a desired task). As our language and/or images seem to represent our environment successfully[9], it follows that accurate predictions can be made by making use of these representations. Thus, the environmental dynamics can be manipulated and/or anticipated with this kind of representations. In other words, if the requirement of accurately mapping the environment to the represen-

[7] A referential representational relationship can be found only in *peripheral* parts of the nervous system. But even in these areas there is no evidence for real stability, as the original stimulus is distorted in the process of *transduction*.

[8] This internal state is the result of the neural system's *recurrent architecture*.

[9] Think about the success of our language, symbolic communication, etc.

tational substratum is satisfied, we do not have to worry about the aspect of generating adequate behavior any more.

From an epistemological and constructivist [29, 30, 48, 73, 64, 74] perspective the claim for an "accurate mapping" is absurd; as has been discussed in section 1, nobody will ever have direct access to the structure of the environment. Hence, it is impossible to determine, how "accurate", "true", or "near" the representation of the environment (be it in our brains or in a scientific theory) compared to the "real" environment is. The only level of accuracy which can be determined is the difference between our own (cognitive) representation of the environment and (our representation of) the (computational) representation which has been constructed by ourselves. In many cases it has turned out, however, that the human representation of the environment is not the best solution to a given problem – consequently, it is questionable to elevate our own representation of the environment (and the resulting representational categories) above other forms of representation and to use them as a standard against which other forms of representation have to compete. It is by no means clear why our (cognitive or even scientific) representation of the world should be more accurate or more adequate than any other form of representation which is capable of accomplishing the same task!

From the previous section follows that there is no empirical evidence for explicit propositional or "picture-like" representations in the brain. This implies that neural systems do *not* generate – obviously quite – adequate behavior by making use of referential representations. It can be shown that any natural nervous system is the result of a long *phylo-* and *ontogenetic* process of *adaptation* and *development*. The goal of this process is *not* to create an accurate model or representation of the environment, but rather to develop these physical (body and representational/neural) structures which embody a (recurrent) transformation being capable of *generating functionally fitting (i.e., successful) behavior*. In natural (cognitive) systems it seems that the aspect of generating behavior is more important than the aspect of developing an accurate model of the environment. What we can learn from these systems and their *adaptational strategies* is that it is *not* necessary to possess an accurate mapping/representation of the environment in order to generate successful behavior. As "accurate representation" of the environment means "accurate" compared to our own representation of the environment, it does not follow necessarily that an "inaccurate" representation is not capable of producing more efficient behavior.

2.2.4 Explicit representation

Both approaches have in common that they are based on an *explicit* representation of the environment. I.e., in propositional models one finds symbols, rules, semantic networks, etc. which explicitly refer to certain states of the environment. In the case of mental imagery explicit visual categories (e.g., mental images, cognitive maps, icons, etc.) are assumed to refer to the environment. Both forms of representation are "accurate" as far as their referential character is concerned. As they match at least one of our representational categories (e.g., language or [mental] images), they can be said to be accurate, if their structure mirrors (our representation of) the environment to a certain degree. In other words, they are only accurate in the context of our own representational categories.

Semantic transparency (in the sense of [15]) is present in both cases. I.e., each representational entity can be assigned a "semantic value" (e.g., the environmental phenomenon it refers to). This kind of representation seems to be based on the concepts which can be found in the von Neumann machine: there are variables whose values refer explicitly to certain environmental states. Again, from a neuroscientific perspective, the concept of semantic transparency seems to be rather rare in the case of natural neural systems (e.g., distributed representation, [26, 65, 66]). If at all, this kind of representation can be found in peripheral parts, where it is possible to assign certain semantics to neural activations, as an observer can correlate them with environmental events. As soon as recurrent activations are involved, it becomes almost impossible to determine the semantics of activations or activation patterns. Although the concept of semantic transparency and of a stable referential relationship has to be given up, natural as well as artificial neural systems are performing extremely well. We can take this as

further evidence for the hypothesis that an accurate mapping representation is not necessarily the most important ingredient for successfully accomplishing a certain task.

2.2.5 Externalized "human" representational categories

It is clear that both, pictorial and propositional representations are the result of complex neural processes which lead to a certain behavior. These behaviors are externalized or these representations are internally experienced as language or pictures. Consequently, whenever we are speaking about propositional and/or pictorial representations we are not dealing with representations which are used by neural systems, but rather with *results* of complex dynamic processes which are making use of the more fundamental neural representations. Thus, it seems that the level of behavioral observations (of linguistic or pictorial representations) is confused with the level of generating these representations in the propositional/pictorial approach. Nothing justifies the assumption that the dynamics being responsible for generating pictorial/propositional representations also makes use of these representational categories in order to generate them.

As has been discussed, neither empirical nor epistemological evidence can be found which supports such an assumption. In fact, it turns out that neural representations are not only based on a different substratum, but also on a completely different concept of representation. This view of representation is based on adaptive processes and on the concepts of system relativity and functional fitness [29, 57, 59]. They do not fit very well into the referential concepts of our traditional understanding of representation. Our neural system *constructs* these (referential) representational categories only in order to "simulate" and give us some kind of "cognitive stability" which gives rise to phenomena, such as more or less successful language, communication, science, etc. Looking more closely at these phenomena, we realize that they are not as stable as they might appear at first glance: the meaning of symbols is shifting over time, the scientific concepts and claims of "truth" and "objectivity" are not as appropriate [19, 21, 20, 47, 28][10] as many would wish, etc.

Our representational domain seems to be more dynamic and plastic than we are aware of. As an implication of these considerations it is necessary to question these traditional concepts of representation – linguistic and propositional representations are only a very crude and misleading way to characterize the representational processes going on in our brains. Using these externalized representations as a basis for an explanation of internal representations is a *methodologically* extremely *questionable* procedure. Projecting these traditional "external" representational categories to neural systems could be an explanation to why we have so many difficulties with interpreting what is going on in natural and artificial neural systems. I.e., no match between our pictorial or propositional representations and the neural representational categories can be found.

2.2.6 "Designer solutions" and projection (methodological problems)

From the previous section follows that most approaches in AI and traditional cognitive science turn out to be "designer solutions"; i.e., instead of studying the structure and dynamics of internal neural representations and how they acquire knowledge from the environment, externalized (linguistic or pictorial) representations are *projected back* into the representation mechanisms of the cognitive model. In other words, an external human observer/designer projects his/her own (linguistic, pictorial, etc.) representation of the world into the observed organism and into the cognitive model which he/she is constructing. This implies that the resulting representational system corresponds partially with the designer's own view, representation, and interpretation of the world.

A comparison with (natural) neural systems shows that their representational structure is *not* the result of a projection of already "prefabricated" and pre-represented (propositional or pictorial) representations, but rather of a long history of *phylo-* and *ontogenetic processes of adaptation.* Representations have not been projected and/or somehow transferred into the neural representa-

[10] Think, for instance, of the history of science, the shifts of scientific paradigms [43], etc.

tion system, but have *developed* in an active process of interaction with the environment. Both the genetic material and the neural structure (as well as the attached body systems) are crucial for the representational function of a natural cognitive system. The structure of the neural system itself *embodies* the knowledge which has "accumulated" in the course of this history. The process of adaptation takes place individually – this implies that the representational structure and categories may vary even within a single species.

2.2.7 Processing

In both representational approaches a similar concept of *processing* is applied: an algorithm manipulates/operates on the representational structure (i.e., on the symbols or mental images). There is a clear *distinction* between the processing part and the representational entities, on which these processes operate (i.e., processor-memory distinction). The processing is actively involved in the dynamics of the system, as it operates on the representations. The representations, on the other hand, seem to play a rather passive role for two reasons: (a) as mentioned above, they are the result of having been projected from the human representation system to the artificial representation system (i.e., they are passive in the sense of being preprocessed and passively mapped); (b) an algorithm executes operations over these representations (i.e., they remain rather passively as they are manipulated by the algorithm).

This concept of distinguishing between processing and memory has its roots in the structure of the Turing machine which inspired the whole computer metaphor for cognitive processes. In neural systems, however, *no* such distinction can be found. Normally the synaptic connections/weights are considered to "hold the knowledge" of the neural system. It is not clear which part of the system takes over the role of the processor. Furthermore, the synaptic weights (i.e., the neural system's "knowledge") turn out to be *not passive* at all – they are responsible for controlling the flow/spreading of the patterns of activations. It can be concluded that it is the *interaction* between the patterns of activations and the configuration of the synaptic weights which is responsible for both the representation of the knowledge and for generating the system's behavioral dynamics.

2.2.8 Lack of empirical evidence

As we have seen in the course of the previous sections, empirical/neuroscientific evidence for the propositional as well as pictorial approach is rather poor. Of course, there are areas in the brain which seem to be related to the processing of language, semantics, propositions, mental images, etc. – the only thing which is known from these areas is that, if they are damaged in one way or the other, then certain cognitive abilities are not present any more [37, 14]. Neuroscience provides almost no knowledge or theories concerning the processing mechanisms/architecture underlying these cognitive phenomena. From this poor evidence it seems questionable to postulate representational concepts, such as the pictorial or propositional paradigm.

That is why both approaches restrict themselves to the claim of being a *functionalist* account in most cases; i.e., they describe the functional properties which can be derived from the "behavioral surface" of the observed cognitive system. These behavioral descriptions are used as "explanatory vehicles" for internal representational processes – it is clear that a lot of *speculation* and *common sense concepts* are involved in these explanations/theories about internal representational processes, as the "real" internal/neural structures are never really taken into account. This might have been a valid approach 20 years ago, when neuroscience still had a comparatively poor understanding of cognitive processes. However, with the advent of modern techniques, theories, and methods in empirical neuroscience, as well as of new concepts from *computational neuroscience* ([13, 14, 2, 1, 32, 25, 67] and many others) the picture has changed dramatically; although there is still a far way to go to fully explain "higher cognitive functions" in neuroscientific terms, many basic concepts have been discovered which can be applied to any level of neural processing (e.g., spreading activations, distributed processing and representation, adaptive processes, "Hebbian" learning as the basis for any kind of learning [LTP, LTD, etc.] [33, 10, 54], etc.). Already these findings suggest a completely different concept of (neural) representation mechanisms/concepts than the propositional and/or pictorial approaches do postulate. It seems that the time, in which one can use the excuse that the brain is

too complex to be understood, will come to an end soon.

2.2.9 Evolutionary implausibilities

From an evolutionary perspective it seems rather *implausible* that a cognitive system develops a representational structure which maps its environment as accurately as possible in order to generate successful behavior [56]. The concepts of *adaptation, selection, system relativity*, and *functional fitness* [29, 73] seem to be much more important than the concept of a structural match between the environment and its representation (which is – from an constructivist/epistemological perspective – an absurd goal, anyway). Neural systems are primarily *adaptive systems* which develop in a continuous interaction with the environment; and not in a single process of mapping or projecting the knowledge of a (human) designer to the representational structure. The representation in natural cognitive systems (as well as in artificial neural networks or genetic codes) *incrementally adapts* to the constraints being set by the environment and by the organization of the organism's body- and representation system. It can be shown ([9, 44, 56, 61] and many others [ALife and ANN literature]) that no picture-like, propositional, or referential representation concept is necessary in order to generate behavior which functionally fits into the organism's internal and external environment.

The physical structure of the neural representation system (i.e., its architecture) is altered incrementally on a *trial-&-error* basis. This process is perpetuated and repeated, until some kind of *equilibrium* is reached (e.g., behavior which ensures the organism's survival, a certain task is achieved by minimizing an error, etc.). The interesting point is that the result of this incremental adaptation processes are *not* pictorial or propositional representations in the brain (or the ANN), but rather a *recurrent transformation* being *embodied* in the neural substratum. This transformation is capable of generating behavior, which is necessary for the particular organism's survival, without having to make use of referential representations. It turns out that pictorial or propositional representations are only *one possible* solution to the problem of representation. As can be shown [56, 59], these solutions are highly uneconomical; in most cases this kind of representations require complex processing, memory, etc., mechanisms. In other words, less complex mechanisms would be sufficient for solving the problem of generating functionally fitting behavior. From an evolutionary perspective complex solutions would be rather atypical, since evolutionary processes normally lead to highly economical solutions which make more or less optimal use of the resources which are available. In this sense pictorial or propositional representations turn out to be "luxury solutions" compared to the simplicity of the task and to the simplicity of other (e.g., neural or "adaptive") solutions.

Hence, the requirement of generating functionally fitting behavior is much *less strict* than the requirement of generating successful behavior which is based on a homomorphic, accurate, and referential (e.g., pictorial or propositional) concept of representation.

3 Methodological and epistemological questions

Although most models in cognitive science as well as in human-computer interface development are mainly concerned with technical questions, the following paragraphs will demonstrate that epistemological and methodological considerations in the field of knowledge representation have crucial implications for the structure and success/failure of the model or interface to be developed. The most important problem concerns the question of how we see and experience the environment/world. Whenever one speaks of "*the world*", we have to be aware – at least since I.Kant – that this is impossible *in principle*. As has been discussed, our access to the environment is always indirect; it is mediated by our sensory systems and by the nervous system. Thus, when we speak of "the world", we actually speak of *our representation of the world*. It is the result of a complex process of *construction* which is embodied in our neural structure. Looking more closely, one realizes that this view has to be taken even one step further: when we speak about the world we are not directly externalizing our neural representation of the world, but we rather make use of another representational medium, namely language, pictures, icons, etc. Hence, what we are

dealing with, whenever we are communicating, reading a text, etc., is a *second-order representation* (i.e., the representation of the [neural] representation of the world). Of course, language is also represented in neural structures – it is, however, a second-order representation, because it is *embedded* and generated by the (first order) neural representation of the environment[11]. It is used for "describing" these (neural) representations.

As our access to the environment is always mediated by the sensory systems and by the structure of the nervous system, this access is highly *theory-laden* (in the sense of [19, 21, 20, 12]). In other words, any natural sensory system, body system, or nervous system can be interpreted as some kind of "*theory*" about the environment. I.e., all these systems have developed in a complex phylogenetic/evolutionary and ontogenetic process of adaptation and learning – only these organisms have survived (and were capable of reproducing) whose neural/body structures embody a functionally fitting (i.e., viable) knowledge/"theory" about the environment. Think, for instance, of our visual system: the rods and cones in our retina are sensitive to a very small fraction of the whole range of electromagnetic waves [70, 72]. Obviously it has turned out in the course of the evolutionary development that this range of electromagnetic waves holds enough information for maintaining the survival of the human body. Bees, on the other hand, are highly sensitive in the UV-range (where humans are insensitive) – for them it has turned out that this range is important for spotting blossoms[12].

From these simple examples one can see that this neurally- and structurally-*embodied* theory about the environment does *not* depict the environment in the sense that certain body parts or neural entities refer to environmental structures, but they represent a *strategy* of how to *survive* in a specific environment with a specific body structure. Both in the phylo- and ontogenetic case the environmental structure/dynamics does not determine the representational structure, but in the best case *triggers* and *constrains* the development and the function of the neural and body (representation) system. The representation of the environment is actively *constructed* by the dynamics being embodied in the nervous system. From these considerations follows that the representation of the world is always *system-relative* in the sense that it represents a "correct theory" of the world *for a specific organism* with its own specific onto- and phylogenetic history.

This implies that, whenever we are speaking about the environment, we always speak about the *representation of the environment in a specific brain/body* (by making use of a specific form of [second-order] externalization mechanism [e.g., language, pictures, etc.]). Thus, we are always dealing with *one possible interpretation/construction* of the environmental structure which is the result of a specific neural system. These interpretations might differ even within a single species. We can*not* claim that a certain representation/interpretation/theory (even a scientific theory) is "objective", "true", or "ultimate". It is only "true" insofar as it contributes to the survival and the reproduction of the particular organism (i.e., insofar as it is capable of generating functionally fitting behavior). What might represent a "true" theory/representation for one organism, might be "wrong" or a non-viable solution for another. This cannot only be applied to simple organisms from different species, but also to such complex and "objective" processes, such as science (e.g., history of science is full of these examples [43]).

A methodological issue which should be of great interest to the design of cognitive models and human-computer interfaces is the fact that most models are based on *second-order representations.* I.e., the internal representational structure of the model/interface is based on linguistic or pictorial externalizations of humans. It is postulated that these externalizations represent some aspect(s) of the world. From the previous paragraphs follows, however, that these externalized representations represent – if at all – only a fraction of the organism's system-relative internal representation of the world. Whichever artifact we are encountering, it is the externalization of an organism's internal (neural) representation (see also section 1.2.2f). Thus, we are confronted with the result of a long and complex chain of neural pro-

[11] There seems to be a *symbiotic* or even *parasitic* relationship between first and second-order representation.

[12] Flowers are reflecting not only in the (human) visual range, but also in the UV-range. Thus, they show a strong contrast in the UV-range, which "helps" the bees to orient and to find them.

cesses and transformations.

The problem which arises for the design of cognitive models and human-computer interfaces can be characterized as follows: most of these systems are based on propositional or pictorial representations. Although it is postulated that these forms of representation are "internal representations", they are *external second-order observational categories*. I.e., an observer observes the *externalized* linguistic, pictorial, logical, problem solving, etc. *behavior* of a (human) cognitive system and tries to find out regularities and/or patterns in these behavioral actions. By making use of these patterns and of his/her own representational experiences (of the world, of problem solving, etc.) he/she projects these second-order observations/phenomena into the observed organism and postulates that they correspond to the organism's internal representation system (without ever having "opened" and examined the internal structure of this system). In other words, an internal mechanism for generating behavior is postulated without ever having a look at the actual internal mechanism. This is exactly the (methodological) situation in the domain of pictorial and propositional representations.

This implies another problem with propositional or pictorial representations: these external representations are *projected* into the cognitive model and/or human-computer interface. Contrary to natural systems, which are actively acquiring/constructing knowledge in a continuous process of interaction, adaptation, and learning, knowledge is mapped to these artificial systems. I.e., the designer projects his/her pre-represented and pre-processed representations, which themselves are the result of his/her own neural construction processes, to the system where they are used as "*internal* representational structures". In these artificial systems they do not only serve as explanatory vehicles, but also as mechanisms being responsible for generating so-called cognitive phenomena. In other words, the *results* of (natural/neural/cognitive) phenomena (e.g., propositional or pictorial representations) are used for generating cognitive phenomena. In this sense we are dealing with a highly superficial and self-referential view of representation. I.e., externalized cognitive behavioral patterns are postulated to be and used as internal mechanisms for generating exactly these (external) patterns. Instead of projecting these externalized representations to cognitive models and declaring them as internal representations, we should rather look at the internal processes and dynamics of the brain. Only, if we learn more about its internal structures, dynamics, and representational categories, we will be able to create more "successful" cognitive models and "friendlier" human-computer interfaces.

4 Conclusions

The goal of this paper was not to give detailed instructions and solutions for developing more adequate cognitive models and user interfaces. As has become clear from the last sections, I wanted to give reasons why the traditional approaches did not work out as good as originally promised. It turned out that the problems are not for the most part located in the technical domain, but in the epistemological and methodological realm. Although many models of cognition and human-computer interfaces are based on concepts from (cognitive) psychology, traditional cognitive science, or AI, we have seen that their results are questionable. In the course of this paper it has become evident that theories from the disciplines mentioned above postulate a concept of representation which neither corresponds to empirical evidence from (computational) neuroscience nor to epistemological considerations. Rather, it seems that they are both limited by (technical) constraints and by common sense assumptions about representation. Furthermore, in most cases they are based on concepts stemming from computer science. Even models in cognitive psychology (e.g., [3, 75] and many others) seem to be heavily influenced by computer science concepts (e.g., memory-processor distinction, memory as filling a variable, algorithmic [non-parallel, non-distributed] linear processing, etc.).

As an illustration think, for instance, of the endless hierarchies of menues which can be found in a large number of application programs. It is a common problem for users to find themselves a way through these trees and levels. The underlying problem is that human users are *not* "push down automata" with an infinite stack. Instead the application should provide cognitively

adequate tools for navigating in the menu structure (e.g., some kind of graphically represented "map" of the menu structure, display the current options, where to from here, etc.).

The position being suggested in this paper favors an approach which starts from the opposite direction: instead of basing a cognitive model on technical concepts (e.g., automaton theory, generative grammars, etc.), it is proposed to study the structure and dynamics of the actual neural system *first* and only then determine which technology is appropriate to model these findings in order to develop more adequate models of cognition and human-computer interfaces.

References

[1] Anderson, J.A., A. Pellionisz, and E. Rosenfeld (Eds.) (1991). *Neurocomputing 2. Directions of research.* Cambridge, MA: MIT Press.

[2] Anderson, J.A. and E. Rosenfeld (Eds.) (1988). *Neurocomputing. Foundations of research.* Cambridge, MA: MIT Press.

[3] Anderson, J.R. (1990). *Cognitive psychology and its implications* (3rd ed.). New York: W.H. Freeman.

[4] Beakley, B. and P. Ludlow (Eds.) (1992). *The philosophy of mind: classical problems/contemporary issues.* Cambridge, MA: MIT Press.

[5] Bechtel, W. (1988). *Philosophy of science. An overview for cognitive science.* Hillsdale, N.J.: L.Erlbaum.

[6] Becker, A.L. (1991). A short essay on languaging. In F. Steier (Ed.), *Research and reflexivity*, pp. 226–234. London; Newbury Park, CA: SAGE Publishers.

[7] Boden, M.A. (Ed.) (1990). *The Philosophy of Artificial Intelligence.* New York: Oxford University Press.

[8] Boyd, R., P. Gasper, and J.D. Trout (Eds.) (1991). *The philosophy of science.* Cambridge, MA: MIT Press.

[9] Braitenberg, V. (1984). *Vehicles: experiments in synthetic psychology.* Cambridge, MA: MIT Press.

[10] Brown, T.H., A.H. Ganong, E.W. Kariss, and C.L. Keenan (1990). Hebbian synapses: biophysical mechanisms and algorithms. *Annual Review of Neuroscience 13*, 475–511.

[11] Christensen, S.M. and D.R. Turner (Eds.) (1993). *Folk psychology and the philosophy of mind.* Hillsdale, N.J.: L.Erlbaum.

[12] Churchland, P.M. (1991). A deeper unity: some Feyerabendian themes in neurocomputational form. In G. Munevar (Ed.), *Beyond reason: essays on the philosophy of Paul Feyerabend*, pp. 1–23. Dordrecht, Boston: Kluwer Academic Publishers. (reprinted in R.N.Giere (ed.), Cognitive models of science, Minnesota Studies in the Philosophy of Science XV, 1992).

[13] Churchland, P.S., C. Koch, and T.J. Sejnowski (1990). What is computational neuroscience? In E.L. Schwartz (Ed.), *Computational neuroscience.* Cambridge, MA: MIT Press.

[14] Churchland, P.S. and T.J. Sejnowski (1992). *The computational brain.* Cambridge, MA: MIT Press.

[15] Clark, A. (1989). *Microcognition: philosophy, cognitive science, and parallel distributed processing.* Cambridge, MA: MIT Press.

[16] Eckardt, B.v. (1993). *What is cognitive science?* Cambridge, MA: MIT Press.

[17] Eco, U. (1976). *A theory of semiotics.* Bloomington: Indiana University Press.

[18] Eco, U. (1984). *Semiotics and the philosophy of language.* Bloomington: Indiana University Press.

[19] Feyerabend, P.K. (1975). *Against method.* London; New York: Verso.

[20] Feyerabend, P.K. (1981a). *Problems of empiricism. Philosophical papers II*, Volume II. Cambridge; New York: Cambridge University Press.

[21] Feyerabend, P.K. (1981b). *Realism, rationalism, and scientific method. Philosophical papers I*, Volume I. Cambridge; New York: Cambridge University Press.

[22] Fodor, J.A. (1975). *The language of thought.* New York: Crowell.

[23] Fodor, J.A. (1981). *Representations: philosophical essays on the foundations of cognitive science.* Cambridge, MA: MIT Press.

[24] Fodor, J.A. (1990). *A theory of content and other essays.* Cambridge, MA: MIT Press.

[25] Gazzaniga, M.S. (Ed.) (1995). *The cognitive neurosciences.* Cambridge, MA: MIT Press.

[26] Gelder, T.v. (1992). Defining "distributed representation". *Connection Science 4*(3/4), 175–191.

[27] Giere, R.N. (1994). The cognitive structure of scientific theories. *Philosophy of Science 61*, 276–296.

[28] Glasersfeld, E.v. (1983). On the concept of interpretation. *Poetics 12*, 254–274.

[29] Glasersfeld, E.v. (1984). An introduction to radical constructivism. In P. Watzlawick (Ed.), *The invented reality*, pp. 17–40. New York: Norton.

[30] Glasersfeld, E.v. (1995). *Radical constructivism: a way of knowing and learning.* London: Falmer Press.

[31] Goldberg, D.E. (1989). *Genetic algorithms in search, optimization, and machine learning.* Reading, MA: Addison-Wesley.

[32] Hanson, S.J. and C.R. Olson (1990). *Connectionist modeling and brain function: the developing interface.* Cambridge, MA: MIT Press.

[33] Hebb, D.O. (1949). *The organization of behavior; a neuropsychological theory.* New York: Wiley.

[34] Hertz, J., A. Krogh, and R.G. Palmer (1991). *Introduction to the theory of neural computation*, Volume 1 of *Santa Fe Institute studies in the sciences of complexity. Lecture notes.* Redwood City, CA: Addison-Wesley.

[35] Holland, J.H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* Ann Arbor: University of Michigan Press.

[36] Hutchins, E. and B. Hazelhurst (1992). Learning in the cultural process. In C.G. Langton, C. Taylor, J.D. Farmer, and S. Rasmussen (Eds.), *Artificial Life II*, Redwood City, CA, pp. 689–706. Addison-Wesley.

[37] Kandel, E.R., J.H. Schwartz, and T.M. Jessel (Eds.) (1991). *Principles of neural science* (3rd ed.). New York: Elsevier.

[38] Kornblith, H. (Ed.) (1993). *Naturalizing epistemology* (2nd ed.). Cambridge, MA: MIT Press.

[39] Kosslyn, S.M. (1988). Aspects of a cognitive neuroscience of mental imagery. *Science 240*, 1621–1626.

[40] Kosslyn, S.M. (1990). Mental imagery. In D.N. Osherson and H. Lasnik (Eds.), *An invitation to cognitive science*, Volume 2, pp. 73–97. Cambridge, MA: MIT Press.

[41] Kosslyn, S.M. (1994). *Image and brain. The resolution of the imagery debate.* Cambridge, MA: MIT Press.

[42] Kosslyn, S.M. and J.R. Pomerantz (1977). Imagery, propositions, and the form of internal representations. *Cognitive Psychology 9*, 52–76.

[43] Kuhn, T.S. (1970). *The structure of scientific revolutions* (2nd ed.). Chicago: University of Chicago Press.

[44] Langton, C.G. (Ed.) (1994). *Artificial Life.* Cambridge, MA: MIT Press.

[45] Lycan, W.G. (Ed.) (1990). *Mind and cognition. A reader.* Cambridge, MA: B. Blackwell.

[46] Maturana, H.R. (1978). Biologie der Sprache: die Epistemologie der Realität. In H.R. Maturana (Ed.), *Erkennen: die Organisation und Verkörperung von Wirklichkeit*, pp. 236–271. Braunschweig: Vieweg (1982).

[47] Maturana, H.R. (1991). Science and daily life: the ontology of scientific explanations. In F. Steier (Ed.), *Research and reflexivity*, pp. 30–52. London; Newbury Park, CA: SAGE Publishers.

[48] Maturana, H.R. and F.J. Varela (Eds.) (1980). *Autopoiesis and cognition: the realization of the living*, Volume 42 of *Boston studies in the philosophy of science*. Dordrecht; Boston: D.Reidel Pub. Co.

[49] McClelland, J.L. and D.E. Rumelhart (Eds.) (1986). *Parallel Distributed Processing: explorations in the microstructure of cognition. Psychological and biological models*, Volume II. Cambridge, MA: MIT Press.

[50] Mitchell, M. and S. Forrest (1994). Genetic algorithms and artificial life. *Artficial Life 1*(3), 267–291.

[51] Newell, A. (1980). Physical symbol systems. *Cognitive Science 4*, 135–183.

[52] Newell, A., P.S. Rosenbloom, and J.E. Laird (1989). Symbolic architectures for cognition. In M.I. Posner (Ed.), *Foundations of cognitive science*, pp. 93–131. Cambridge, MA: MIT Press.

[53] Newell, A. and H.A. Simon (1976). Computer science as empirical inquiry: symbols and search. *Communications of the Assoc. for Computing Machinery (ACM) 19*(3), 113–126. (reprinted in M.Boden (ed.), The Philosophy of Artificial Intelligence, Oxford University Press, 1990; in German in D.Münch (ed.), Kognitionswissenschaft, Suhrkamp, 1992).

[54] Nicoll, R.A., J.A. Kauer, and R.C. Malenka (1988). The current excitement in long-term potentiation. *Neuron 1*(2), 97–103.

[55] Osherson, D.N. and H. Lasnik (Eds.) (1990). *An Invitation to cognitive science*, Volume 1-3. Cambridge, MA: MIT Press.

[56] Peschl, M.F. (1993). Knowledge representation in cognitive systems and science. In search of a new foundation for philosophy of science from a neurocomputational and evolutionary perspective of cognition. *Journal of Social and Evolutionary Systems 16*, 181–213.

[57] Peschl, M.F. (1994a). Autonomy vs. environmental dependency in neural knowledge representation. In R. Brooks and P. Maes (Eds.), *Artificial Life IV*, Cambridge, MA, pp. 417–423. MIT Press.

[58] Peschl, M.F. (1994b). Embodiment of knowledge in the sensory system and its contribution to sensorimotor integration. The role of sensors in representational and epistemological issues. In P. Gaussier and J.D. Nicoud (Eds.), *From perception to action conference*, Los Alamitos, CA, pp. 444–447. IEEE Society Press.

[59] Peschl, M.F. (1994c). *Repräsentation und Konstruktion. Kognitions- und neuroinformtische Konzepte als Grundlage einer naturalisierten Epistemologie und Wissenschaftstheorie.* Braunschweig/Wiesbaden: Vieweg.

[60] Polanyi, M. (1966). *The tacit dimension.* Garden City, N.Y.: Doubleday.

[61] Port, R. and T.v. Gelder (Eds.) (1995). *Mind as motion: explorations in the dynamics of cognition.* Cambridge, MA: MIT Press.

[62] Posner, M.I. (Ed.) (1989). *Foundations of cognitive science.* Cambridge, MA: MIT Press.

[63] Roth, G. (1991). Die Konstitution von Bedeutung im Gehirn. In S.J. Schmidt (Ed.), *Gedächtnis*, pp. 360–370. Frankfurt/M.: Suhrkamp.

[64] Roth, G. (1994). *Das Gehirn und seine Wirklichkeit. Kognitive Neurobiologie und ihre philosophischen Konsequenzen.* Frankfurt/M.: Suhrkamp.

[65] Rumelhart, D.E. and J.L. McClelland (Eds.) (1986). *Parallel Distributed Processing: explorations in the microstructure of cognition. Foundations*, Volume I. Cambridge, MA: MIT Press.

[66] Rumelhart, D.E., P. Smolensky, J.L. McClelland, and G.E. Hinton (1986). Schemata and sequential thought processes in PDP models. In J.L. McClelland and D.E. Rumelhart (Eds.), *Parallel Distributed Processing: explorations in the microstructure of cognition. Psychological and biological models*, Volume II, pp. 7–57. Cambridge, MA: MIT Press.

[67] Sejnowski, T.J., C. Koch, and P.S. Churchland (1990). Computational neuroscience. In

S.J. Hanson and C.R. Olson (Eds.), *Connectionist modeling and brain function: the developing interface*, pp. 5–35. Cambridge, MA: MIT Press.

[68] Shepard, R. and J. Metzler (1971). Mental rotation of three-dimensional objects. *Science 171*(972), 701–703.

[69] Shepherd, G.M. (Ed.) (1990). *The Synaptic organization of the brain* (3rd ed.). New York: Oxford University Press.

[70] Sterling, P. (1990). Retina. In G.M. Shepherd (Ed.), *The Synaptic organization of the brain* (3rd ed.)., pp. 170–213. New York: Oxford University Press.

[71] Stillings, N.A., M.H. Feinstein, and J.L. Garfield (Eds.) (1987). *Cognitive science: an introduction.* Cambridge, MA: MIT Press.

[72] Tessier-Lavigne, M. (1991). Phototransduction and information processing in the retina. In E.R. Kandel, J.H. Schwartz, and T.M. Jessel (Eds.), *Principles of neural science* (3rd ed.)., pp. 400–419. New York: Elsevier.

[73] Varela, F.J., E. Thompson, and E. Rosch (1991). *The embodied mind: cognitive science and human experience.* Cambridge, MA: MIT Press.

[74] Watzlawick, P. (Ed.) (1984). *The invented reality.* New York: Norton.

[75] Winston, P.H. (1992). *Artificial Intelligence* (3rd ed.). Reading, MA: Addison-Wesley.

Knowledge Objects

Xindong Wu, Sita Ramakrishnan, Heinz Schmidt
Department of Software Development
Monash University
900 Dandenong Road
Melbourne, VIC 3145, Australia
E-mail: {xindong,sitar,hws}@insect.sd.monash.edu.au

Keywords: AI programming, rules, objects, intelligent objects, knowledge objects

Edited by: Matjaž Gams
Received: May 15, 1995 **Revised:** October 25, 1995 **Accepted:** November 28, 1995

True improvements in large computer systems always come through their engineering devices. In AI, one of the fundamental differences from conventional computer science (such as software engineering and database technology) is its own established programming methodology. Rule-based programming has been dominant for AI research and applications. However, there are a number of inherent engineering problems with existing rule-based programming systems and tools. Most notably, they are inefficient in structural representation, and rules in general lack software engineering devices to make them a viable choice for large programs. Many researchers have therefore begun to integrate the rule-based paradigm with object-oriented programming, which has its engineering strength in these areas. This paper establishes the concepts of knowledge objects *and* intelligent objects *based on the integration of rules and objects, and outlines an extended object model and an on-going project of the authors' design along this direction.*

1 Introduction

Artificial intelligence (AI) is a subject concerned with the problem of how to make machines perform such tasks, like vision, planning and diagnosis, that usually need human intelligence and are generally difficult to be carried out with conventional computer science technology. AI problems are normally NP (non-polynomial) hard by nature. Different from conventional numerical computations, AI research has concentrated on the development of symbolic and heuristic methods to solve complex problems efficiently. Since the 1980's, AI has found wide realistic applications in those areas where symbolic and heuristic computations are necessary. For example, expert systems have produced startling economic impact.

Because of the need for symbolic and heuristic computation, AI has its own programming methodology [Wu 94b], and rule-based programming has been dominant in AI research and applications. It is probably an axiom of AI that domain expertise is always rule-governed. Firstly, even in the world at large, people have a tendency to associate domain expertise with regularities in behaviour and often explain behaviour by appealing to such regularities. Secondly, knowledge in an AI system often depends on some domain expert(s)' heuristics, which can be easily and naturally encoded into the "IF ... THEN" structure. Therefore, rule-based systems have become one of the most widely used models of knowledge representation in AI, in particular expert systems. Rather than expressing logic calculus about the world as in Prolog-like logic programming systems or computing the numeric values defined over data as in conventional programming, rule-based production systems normally determine how the symbol structures that represent the current state of the problem should be manipulated to bring the representation closer to a solution. Problems that have been solved in production systems can be usually encoded in LISP or PROLOG, of course;

the point is that production systems and rule-based programming languages are specifically designed to solve those problems, and as a result they solve those problems rather well. Meanwhile, rules are the essential component for both rule-based production systems (or rule-based systems) and logic programming systems. Since heuristic knowledge is of major concern in this paper, rule-based programming is more production systems oriented. However, if you do not plan to deal with inexact rules, you can use logic programming to replace rule-based programming hereafter.

Rule-based programming has many advantages, such as uniformity and naturalness, but there are also several significant disadvantages inherent in the mechanism:

1. Rules are inefficient in structural representation. Encapsulation of all relevant information of a single entity is hard with rule-based programming.

2. Rules in general lack software engineering devices such as modules, information hiding, and reuse to make them a viable choice for large programs.

3. It is as yet unclear how large sets of rules are best partitioned and distributed in networks of multicomputers in the interest of collaborative knowledge systems, parallel reasoning, partial knowledge or dynamic knowledge (re)configuration.

To avoid these engineering problems, many researchers have begun to integrate the rule-based paradigm with object-oriented programming, a powerful technology from software engineering and the database community. Section 2 outlines the main features of object technology. Section 3 discusses two different ways for the integration of objects and rules. Section 4 explores the idea of intelligent objects by describing an extended object model with two layers of constraints and elaborates these notions with an aircrew scheduling example. Section 5 defines knowledge objects and introduces the design of an on-going project at Monash University.

2 Object Technology

Object technology in software engineering makes it easier to develop, maintain and reuse a wide range of applications. These applications are mainly concerned with data processing. Object orientation attempts to model the behaviour patterns of collections of cooperating physical entities in the real world. Object-oriented programming (OOP) provides a better way of defining data and procedures that are associated with these physical entities than conventional imperative languages such as C, Pascal and Fortran.

OOP was first discussed in the late 1960's when the so called "software crisis" began in large systems development. Methods have evolved since then and have shifted the emphasis from a problem of coding to object-oriented design (OOD). The primary aim of OOD is to improve productivity, increase quality and elevate the maintainability of large software systems [Coad & Yourdon 91]. The well defined and widely accepted principles are the concepts of the class, encapsulation, inheritance and polymorphism. At the core of OOD is the class which represents a real world entity by grouping all of its data attributes and procedural operations together into a neatly encapsulated package.

Software productivity is improved primarily by reducing the amount of time required for detecting and removing defects from programming code. Reusing software, in the form of "class libraries" can produce startling increases in productivity and greatly reduce the amount of errors in a large program. However, the emphasis on productivity could have obscured the need for improvements in software quality. Processes that produce high-quality products early in development, such as analysis and design, can greatly reduce errors discovered later in development such as coding and testing and can dramatically improve software quality [Coad & Yourdon 91]. Maintainability, the final objective of OOD, is accomplished by separating the dynamic parts of a system from those parts which are stable. A robust system must be designed with the expectation of change according to the ever changing requirements of clients. Achieving all of these objectives together in a single system is always difficult to accomplish and more than often a trade-off is necessary. With appropriate use, however, the

principles of OOD will assist in achieving these goals.

2.1 Abstraction and encapsulation

Abstraction is the principle of capturing useful information by ignoring all the detailed features of an entity that are not relevant to understanding what it does or what it is. Rather than trying to comprehend everything about the entity all at once, we select only part of it.

Abstraction consists of "data abstraction" and "procedural abstraction". Procedural abstraction can already be found in most imperative programming languages in the form of functions and procedures, which can be used to reduce the complexity of programming code. In OOD, data abstraction is carried out by the definitions of abstract data types (ADTs) – commonly called *classes* or *types* [Atkinson *et al.* 92]. An ADT is defined in terms of data items and the operations, called *methods* in OOP, that can be applied to these data items. The data within the ADT can only be modified and manipulated by these methods. The resulting notion of encapsulation leads to a separation of interface and implementation. The data of an ADT can only be accessed via the specified interface, while the implementation details such as the operations are hidden and at the discretion of the class implementor or the dynamic decisions of the ADT.

By encapsulation in OOD, each component of a program should hide a single design decision. The interface to each module should be designed so as to reveal as little as possible about its internal implementation details. A language which provides this feature enables the designer to keep related components of a program together in the form of a package in the hope that later changes can be carried out within this package.

2.2 Classes

When using an ADT in an imperative language with constructs such as *records* (used in Pascal) or *structs* (used in C) the designer will normally create routines which manipulate the structure. Operations or routines defined for the data structure in one construct cannot be used for another structure in these languages. In an object-oriented programming language (OOPL) such as C++, the data structure and the operations are bound together into one package, called a *class*. A class can have private and public data. The private data cannot be seen or modified by the user without using the public interface, known as *member functions* in C++. This prevents accidental modification of the data and improves code quality by reducing the amount of bugs evident in the code [Eckel 93].

Variables or instances of a class are called *objects*. There is a fundamental difference between an object and a class: the class is the definition for the data structure, and an object is an instance of that data structure. More than one object can be created from a class definition. This distinction has also led to distinguish *object-based* programming languages from *object-oriented* (OO) ones. In OO languages, objects encapsulate a concrete data structure and a behavior, and they do not need a type. In OO languages, objects are classified and all objects of the same class share the same behavior. Therefore in an OO language the procedures and functions defined on a data structure are described as part of a class and the class is considered as a generic device for instantiating objects. In this case, the user can use the member functions provided by the public interface of a class to pass messages to objects of the class, and the objects control their own actions and can remember their current state.

Two special member functions, called *constructors* and *destructors* in C++, are provided in many OO languages to allow the user to pass a message to a class and create or destruct an object. A constructor is used to initialise an instance of a class by allocating memory for an array, for instance. Destructors on the other hand are used for clean-up operations, such as freeing any memory the object may have been explicitly allocated.

2.3 Inheritance

In an OOPL a user-defined class can inherit features of another, thus promoting a much higher level of code *re-use*. Inheritance allows a designer to specify common attributes and services in one class, and then specialise and extend those attributes and services into specific cases. One class may also inherit the properties of more than one

other class, and this is called *multiple inheritance.*

Single and multiple inheritance is supported by C++ by means of *derived* classes. A derived class is declared by following its name with the names of its *base* classes. A derived class can inherit either the base classes' public parts or both their private and public parts. This is still an issue left open-ended, to be used at the discretion of the designers of individual OOPLs. To support multiple inheritance the derived class may form a base class of another derived class permitting the construction of class hierarchies. An inheritance structure is one of the ways of offering reusability, extendibility, lower maintenance cost and of achieving the software engineering goals that designers have been aiming at for 20-30 years [Henderson-Sellers 92].

2.4 Polymorphism and dynamic/late binding

Although not everyone in the OO community has agreed on it, *polymorphism* is one of the most powerful concepts of OOD. It is the concept of sending a message from one object to other objects in an inheritance hierarchy and invoking the most appropriate behaviour for the object. Polymorphism presents the property of *operator overloading.* In C++ overloading allows the user to specify member functions with the same name which perform different functions according to how many, and the types of parameters passed.

To allow the functionality of polymorphism the compiler of an OOPL cannot bind the operation names to programs at compile time [Atkinson *et al.* 92]. Therefore, operation names must be resolved at run-time. This delayed translation is known as *late* or *dynamic binding.* Similarly, overloading allows the same member function to be declared for all of the different derived classes of shape. Polymorphism and dynamic-binding are provided in C++ through the use of *virtual member functions.* A virtual function is provided with definition in its base class, but may be redefined in derived classes. That is, a virtual function may have different versions in different derived classes and it is the responsibility of the run-time system to find the appropriate version for each call of the virtual function. Functions that are not marked as virtual may be bound statically to the base class in which it is defined which allows for easier implementation.

There are also other forms of polymorphism than the dynamic binding described above, most of which are available in OOP but also in other languages. *Parametric polymorphism* refers to functions that work in the same way on many different data structures, such as *append* works on lists of small integers and also lists of arrays. Such polymorphism is described by parameterised classes or – in C++ – templates. *Ad hoc* polymorphism is the concept of syntactic or "sugar" overloading, where a programmer introduces some ambiguous notation that is statically resolved by a compiler, for instance by considering the number of parameters. When we wish to distinguish the message passing polymorphism in OO, we speak of *subtype polymorphism.* This terminology suggests that a derived class introduces a subtype of its base class: The instances or objects of the derived class can always be considered as instances of the base class, because all messages for the base class are understood and operated according to the abstraction of the base class.

2.5 Object-oriented databases

Another area where object technology has also found wide interest is object-oriented database systems [Cattell *et al.* 91]. In object-oriented database systems, complex data structures (e.g. multimedia data) can be defined in terms of objects. Data that might span many tuples in a relational DBMS can be represented and manipulated as a data object. Procedures/operations as well as data types can be stored with a set of structural built-in objects[1] and those procedures can be used as methods to encapsulate object semantics. Containment relationships between objects may be used to define composite or complex objects from atomic objects. An object can be assigned a unique identifier which is equivalent to a primary key in a relation. Relationships between objects can also be represented more efficiently in object-oriented data models by using a more convenient syntax than relational joins. Also, most object-oriented DBMSs have type inheritance and

[1]These built-in objects are what we call classes in Section 2.2. More than often, the term objects is used for both classes and objects in the literature including the rest of this paper, when the distinction between classes and objects is not emphasized.

version management as well as most of the important features of conventional DBMSs. The mandatory features of an object-oriented database, as presented by [Atkinson *et al.* 92], extend the basic set of OOD principles to include persistence, versioning and integrity control.

Objects in OOP and OODBs are similar in that they require abstraction, inheritance and polymorphism, but there are several important differences. First, database objects must *persist* beyond the lifetime of the program creating them. Second, many database applications require the capability to create and access multiple *versions* of an object. Third, highly active databases, such as those used for air traffic control and power distribution management, require the ability to associate *conditions* and *actions* where the actions are triggered when the constraints are satisfied. Finally, database integrity control demands the capability to associate *constraints* with objects.

2.6 Objects vs. frames

Objects are in many ways similar to the frame structure which was first developed in mid-1970's [Minsky 85] and has found wide use in AI and other knowledge based application systems. A frame is a static data structure used to represent well-understood, stereotyped situations. It organises our knowledge of the world based on past experiences. We can revise the details of these past experiences to represent the individual differences for new situations. A frame includes declarative and procedural information in predefined internal relations. The internal relations reflect the semantic knowledge of the specific entity corresponding to the frame. Clearly, any object can be viewed as a specific frame.

Frames make it easier to organise knowledge hierarchically. We can describe in a frame an object with its various attributes and other relevant objects and think of the frame as a single entity for some purposes and only consider details of its internal structure for other purposes. Procedural attachment is a particularly important feature. We use procedural attachment to create *demons*, which are procedures that are invoked as a side effect of some other action in the overall system.

Objects and frames both have identifiers (or names) and hierarchies, and both have procedures associated with the data slots. Both permit single and multiple data inheritance. However, there are also clear differences between the two technologies. Firstly, the procedures of frames, demons, are not directly activated by the programmer, rather they are activated by the situation, i.e., when a data slot is accessed, updated or deleted. Procedural attachments of frames might be defined that automatically perform certain tasks, such as finding an attribute value when none exists, or making sure related attributes are updated when one or the other is changed. This passive structure is in contrast to the methods of OOP that are directly activated by the programmer by message passing. The procedural methods in an object actively respond to messages received from other objects. Also, polymorphism is not offered by frames although one can argue that it could be implemented.

Secondly, a composite frame can contain pointers to other (primitive and/or composite) frames in its slots, and the other frames do not have to be in a specific hierarchy. This is not allowed in class based OO languages. An object can only inherit data and methods from the classes of its higher hierarchy. Frames and objects both permit single and multiple data inheritance.

Finally, frames also differ from objects in their openness. They are designed to work with an inference engine, and their attributes are always open for interaction with any and all pattern-matching rules. This is in contrast to pure objects in which the attributes and methods are so tightly encapsulated, you cannot tell which is which from the outside. Furthermore, the private data in objects cannot be seen by the user.

Objects are a full programming system, designed as much for encoding procedures as data. Frames were never designed to be a full programming system by themselves. Information hiding is a key for objects, and the source of much of the maintainability of object-oriented applications. However, frames have to be open to the inference engine, so whenever any data changes, it knows what rules to activate.

3 Integration of Objects and Rules

It is hard to say whether rule-based programming or OO languages are superior in computational strength. Rule-based programming expresses relationships between objects very explicitly. However, they don't express updates clearly. OO programming is weak in inference power due to its procedural origin, but updates are defined clearly by assignments. It has the central ideas of encapsulation and reuse which encourage modular program development.

On one hand, while the OO paradigm provides efficient facilities for encapsulation and reuse, it does not support inference engines for symbolic and heuristic computation. A clear advantage of rule-based programming is that recursion can be easily defined within rules while difficult in objects. On the other hand, rule-based programming is very limited in structural representation and for large systems. Therefore, it would be very useful if we can integrate both of them in a seamless and natural way in order to exploit their synergism. It seems as if objects and rules are made for each other. Objects are the best way to simulate or model a problem domain. Rules can be designed to capture and encode human expertise that is applied to a problem domain. A natural way seems to be use objects for modeling the domain and rules to represent decision-making applied to the domain.

The two paradigms are both self-important and it is not appropriate to say that one should be the master and the other the slave in general, but depending on the application domains, choosing one of them as the basis and building the other on the top are necessary given that a seamless integration is not yet available and constructing one may well be very time consuming.

3.1 Incorporating rules into objects

It is argued in [Wong 90] that it is undesirable to implement objects within rule-based programming, since rule-based programming is not as portable as OO programming. One way to get round this is to implement rules within objects. In Prolog++ [Moss 94], for example, an object layer is designed as an emcompassing layer for Prolog rules. In this paradigm, objects can call Prolog rules without any special annotation, and if a Prolog predicate is redefined within the Prolog++ class hierarchy, the definition will be taken by default. Rules can be used to make an object's semantics explicit and visible [Graham 93, Zhao 94]. They can also provide heuristic procedural attachment in methods. Actually methods within objects can always be implemented in the form of rules.

Rules can be defined in an independent rule base so that the methods in objects can call the corresponding predicates (rule heads), in the form of, e.g., obey statements in [Wong 90]. We can of course implement a set of rules with the same rule head in the form of objects, such as the rule objects and reasoner object/class in Section 4, although some of the OO advantages like inheritance, cannot be found from such objects.

Rules within objects can be divided into two categories [Odell 93]: constraint rules and derivation rules. The former define restrictions of object structure and behavior, such as consistency and constraints, and the latter are used to infer new data from existing data. In [Kwok & Norrie 94], for example, an object has four protocol parts: attributes, class methods, instance methods and rules. Rules can be activated by messages as methods.

3.2 Embedding objects into rules

In a rule-based system, data in the working memory (or database) represents the state of the system and is used to fire rules. In an OO system, the state is characterised by the the data items in objects. Therefore, a natural integration of objects and rules is to use objects as storage for the working memory in a rule-based system, and rules execute actions depending on the values of objects in the working memory. A number of AI tools such as CLIPS [Giarratano 93] have provided such facilities to embed objects in rules.

An alternative way is use OO languages as the basis and implement rules which describe relationships of objects on the top of them. Domain expertise always relates to inter-relationships between objects, therefore a declarative query language for expressing these inter-relationships is very useful in integrated systems. This is the approach adopted by Ramakrishnan (1993) and is discussed in detail in the next section.

4 An Extended Object Model

4.1 Intelligent objects

An object which must satisfy dynamic constraints is referred to as an *intelligent object*. An intelligent object is "intelligent about the context" in which the object interacts with a rule base. In this approach, the static rules that must be satisfied by the methods of an object are embedded within the object using the OO language facilities and the dynamic rules of the intelligent object are available from the rule base component. In this cooperative way, integration of rules and objects is built using a loosely coupled component based architecture made up of domain application objects, a rule base cluster and an inference cluster [Ramakrishnan 94c]. The next subsection shows how an object model of a class based OO language such as Eiffel [Meyer 92] can be extended to support two layers of constraints. The rest of the section gives a practical example using this extended object model in Eiffel.

4.2 Layers of constraints

A conceptual schema describes the syntactic information structure and the semantic constraints that exist in an enterprise. The information structure should reflect the pattern in the real world and OO languages such as Eiffel [Meyer 92] can be used to specify this pattern as static class descriptions.

A class description defines the behaviours of its instantiated objects. In the Eiffel language, the constraints that must be satisfied as part of the method invocation can be spelt out as assertions. The static integrity constraints are called class *invariants*. These represent formulas that hold true for the corresponding objects in all possible (observable) circumstances. Each method can be further constrained by *preconditions* and *postconditions*. Hoare logic [Hoare 89] forms a solid basis for the informal notion of "design-by-contract" [Meyer 89]. It can be shown that local compliance with such assertions implies global correctness and indeed stability, i.e., internal changes of a class that are correct relative to its interface cannot affect global correctness [Schmidt & Zimmermann 94]. This includes a kind of *superclass encapsulation* because subclasses cannot be affected even if the changed methods are inherited.

How do these constraints work? They can be used to specify the contractual agreement between the user of the behaviour and the provider of the behaviour. The user is responsible to satisfy the preconditions, the object is assumed to guarantee its invariant, and the method then, correctly implemented, must terminate by delivering the postcondition and reestablishing the invariant. In this way the class hierarchies can be viewed as layers of constraints that enforce the requirements of behaviour specifications.

Some systems require their business rules and regulations to be captured and available for scrutiny by government authorities. Such systems could benefit from the inclusion of explicit rules to control the behaviour of objects dynamically and should be considered explicitly in the analysis, design and implementation models [Ramakrishnan 94b]. For example, business rules could express dynamic constraints that must be met by objects that are required to satisfy these business rules. These dynamic constraints form the second layer of constraints on top of the first layer of static constraints. The two-layered constraint model of an object expresses the mechanisms by which incremental evolution of a system can incorporate business rules [Ramakrishnan 94a]. The dynamic constraints that are required to specify these business rules are specified declaratively and implemented as a separate component by reusing the parsing library abstractions available in Eiffel and building an attribute grammar to describe the rules [Ramakrishnan 93]. The declarative nature of these rules promotes user friendly interaction with the system and enables ease of evolution of the business rules and regulations.

4.3 An aircrew scheduling example

An aircrew scheduling example in this subsection is used to discuss the proposed model. The planners involved in aircrew scheduling must satisfy the business rules or constraints prior to the allocation of crews to flights. Some of this domain knowledge can be captured and represented as rules to be considered in the allocation process. Other constraints such as a last minute change to the availability of an aircrew have to be handled

by the planners online as part of the interactive scheduling system.

The business rules are expressed as production rules which have been widely adopted in knowledge-based systems. The two main components of a knowledge based system are its knowledge base which is a repository of production rules in our case and its inference engine [Dillon 93]. In the aircrew scheduling example, the inference engine is a data driven reasoner which uses the rule base to change the state of an application object. The condition of each business rule is coded in the form of *context label: object, attribute, value.* These business rules are represented as a structured document using a simple English language structure shown as follows:

```
setoperating: Given DUTY equals
operating appo dtime maximum is 12.

setpaxing: Given DUTY equals paxing
appo dtime equals to 17.

mixoperandpax: Given DUTY equals
mixoperpax appo dtime maximum is 16
```

These rules have been described using Hedin's [Hedin 89] OO notation for attribute grammars and implemented in Eiffel [Ramakrishnan 93]. These rules form the wrapper layer around application objects and are referred to as *rules using grammar* (RUG). The application objects are stored to capture the conceptual model of the real world. The attributes of these objects reflect the roles played by these objects and as such can be used to trigger the rules in the rule base. The extended object model thus incorporates both objects and rules and are used by those application objects that interact with the business rules. These interactions represent the second layer of constraints that these objects have to satisfy. The model proposed here is shown in Figure 1 in which the component marked resource allocation jobs (RAJ) includes all the resources and job objects to describe all the entities related to the aircrew scheduling problem. The main feature of the model lies in its ability to treat business rules in a logical and systematic manner so that these rules can also be included as part of the reuse strategy in the incremental evolution of software. In this model, the constraints that may be satisfied by the resources and task objects involved in resource allocation are considered at the following two levels: static type definitions and context related information. Constraints at the first level are specified as part of its class definition. Such a constraint must be satisfied by an object when its behaviourial action(s) are invoked and is specified as assertions in Eiffel [Meyer 92]. The business rules represent the second level of constraints that have to be satisfied by the RAJ objects and are included as a wrapper layer of rules around those objects as shown in Figure 1. The RAJ object participates in the second level of constraint satisfaction by using the context information in the context header of the RUG object. The context header in the business rules represents the role played by the resource object. This second level of constraints is used to activate only those rules which match the active resource rule object and integrates the business rules and an application object in the resource object's constraint satisfaction. Some RAJ objects may not participate in any of these constraint satisfaction schemes, others may participate only at the first level and yet others may have constraints to be satisfied at both levels. The actions (methods, procedures or routines) of the objects have been qualified with either or both of the two levels of constraints, as dictated by the requirements of the objects in their interactions. One of the benefits of using the object-oriented approach is that the semantics of a system can evolve incrementally using the facilities provided by the paradigm for including new methods for various classes (types) over time [VanBiema 90]. The two levels of constraints used in this model, which allow an application object to have these varying levels of constraints, are a powerful additional mechanism through which software may evolve [Barbier 92].

The business rules as shown above have been described in a declarative, simple English-like format. The planners of resource allocation problems can thus encode their rules with ease [Medeiros 91]. The structured document of business rules is reconstructed and semantic actions are applied on the parsed document by collaborating with the lexical and parsing library classes of Eiffel. The language features that are used to describe the syntax and semantics of the RUG rules and the compilation of these rules which generate a parse tree have been discussed elsewhere

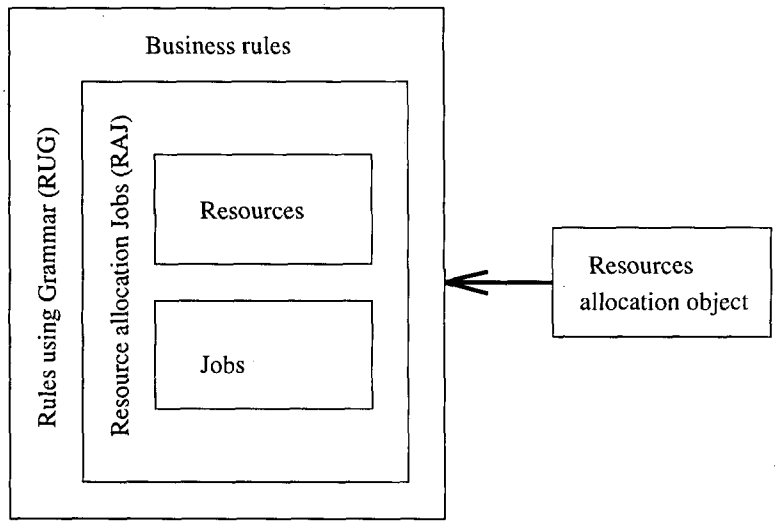


Figure 1: Resource application job objects wrapped with rules using grammar

[Ramakrishnan 93]. The design framework integrates RAJ, which is represented in an OO paradigm, with the rule-based structure of the business rules (RUG) in a single Eiffel language [Meyer & Nerson 90]. The high level architectural design (refer to Figure 2) shows the connection of the major components (clusters). The dynamic model has been used to highlight some communication protocols between certain objects [Rumbaugh *et al.* 91].

4.4 Wrapper layer of rules using grammar

Resource allocation problems require the organisation's business rules to be included as part of the domain model. Business rules may be specified for a number of objects in the application cluster and an object may have to satisfy a number of rules. These rules may contain dependency information between attributes of an object. For example, a `duty` object in the application cluster may contain the following rule: "`If duty is operating then total number of hours that this crew can work is 12 hours.`" The attributes in question are `operating` and `total_number_of_hours`. The `total_number_of_hours` attribute is a derived attribute (calculated) and the rule reflects the condition that must be met in allocating a crew member to a flight as part of their `duty`. These rules or constraints could be specified as assertions (preconditions, postconditions and invariants) in languages such as Eiffel. But, although assertions could be used to specify the constraints that an object and its descendants must satisfy, business rules expressed as a separate component makes them explicit and easy to read and extend. A rule base component cluster should contain rules for resource application objects that can be used as a wrapper layer for objects in the application objects cluster (refer to Figure 1). The wrapper must also be satisfied by application objects in addition to their usual constraint rules which can be specified as assertions. The crew allocation process involves interaction between the resource objects. The application objects such as `duty` that have a wrapper layer interact with the rule base component by instantiating a `reasoner` object. The `reasoner` object has access to stored rule base application objects. In the prototype application, the resource object, `aircraft`, has been designed as an object without this semantic wrapper and hence there is no interaction between this object and the rule base component. The `aircraft` object does have to satisfy a postcondition constraint included as part of its definition. But, more explicit business rules could be included as a wrapper in the rule base cluster. Hence, the mechanism for including explicit rules about resource objects is to include the rules for these resource objects in the rule base component and let the control be handled by the inference cluster.

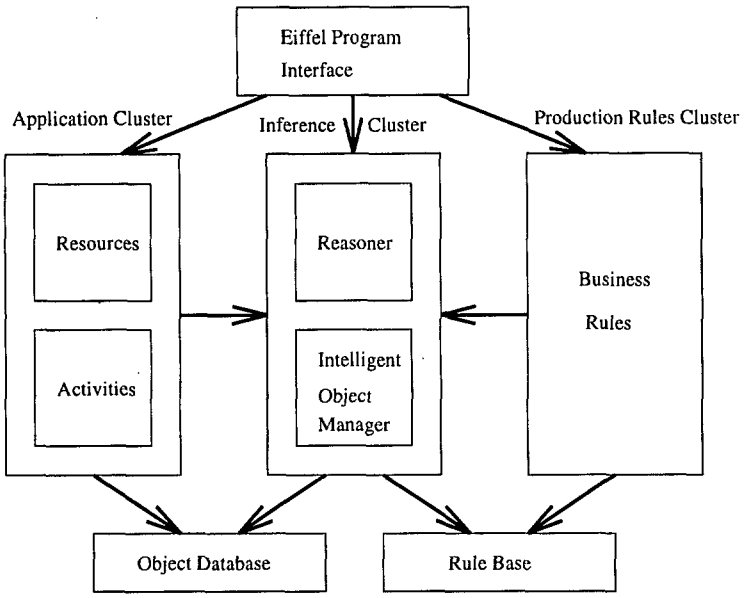


Figure 2: Resource allocation design framework

4.5 Integrating rules and application objects

The object-oriented paradigm provides good techniques for describing taxonomies of objects. But, in traditional OO languages, the order of execution of methods is controlled through the statically defined class hierarchy. These languages do not provide mechanisms to code heuristics explicitly for the order of execution of methods. The methods can be specialised only according to their types through their inheritance relationships and not according to the state of the object.

In our model, the constraints that may be satisfied by the resources and task objects involved in a resource allocation (refer to Figure 1) are achieved through the integration of a rule-based paradigm into the OO language, Eiffel. Rules include a context header which precedes the `if` *condition* `then` *action*. This creates a context sensitive data driven rule-based system which interacts with the application objects in the resource allocation process. The context header may match the messages sent to application objects. The resource allocation data activates only those rules which match the context. This reduces the number of rules to be searched during the allocation process. This observation agrees with Chandrasekaran's observation [Chandrasekaran 92] that viewing knowledge at the appropriate level results in only a subset of the body of knowledge being relevant for consideration, thereby eliminating the need for conflict resolution.

Objects are modelled in terms of their roles or responsibilities. The role is defined by the operations of the object [Jacobson 92]. An operation upon an object is described as part of the definition of a class. A message invokes an operation. The context header represents the role played by a rule, referred to as *a rule object* hereafter. A rule object specifies the action to be taken by the application object as the object's responsibility when the condition is met. A list of valid application objects for a resource allocation system and the responsibilities or roles of these objects are available to the system from the `obj_names` list. Using this central information on valid objects and their roles, action is taken to invoke the appropriate message of the object.

4.6 Constraint satisfaction of business rules

Application objects interact with the reasoner module to check for constraint satisfaction of rules. The reasoner in the aircrew scheduling problem links the application objects to the rule objects, and controls the interaction of the resource objects and the rule objects. The rule objects are retrieved from the rule base. The reasoner controls the rules which are fired by matching the context of the application object against the con-

text header of each rule object. Any extension to the behaviour or role or contextual information enacted by an application object affects the reasoner as well. The new behaviours should be included in the relevant application objects and any new rules added to the rule base to reflect this capability could be fired by adding the appropriate routines to the reasoner.

4.7 Details of the crew allocation process

The crew allocation process involves two levels of constraint satisfaction. When the planner chooses the flights to be included as part of a crew schedule, static constraints are confirmed such as the aircraft scheduled for the flight must have at least two crew members on board. This is an example of an application object with one layer of constraints to be satisfied. A crew is allocated a number of flights to make up their `duty`. The `duty` consists of a number of flights in which the crew is operating the flight in some cases and just traveling as a passenger on the flight in other cases. The attribute of particular interest in a `duty` application object is the derived attribute value for `number_of_operating_hours`. The value for this attribute is calculated by accumulating the `operating_time` of the crew on these flights. It is in the preparation of a crew's `duty` that the business rules are checked and form the second layer of constraints for the `duty` object. This object participates in the second layer of constraints by creating a reasoner object which in turn activates the rules in the rule base. The `duty` object participates in a number of roles during its life time. The current role enacted by this object would set its context and a match on this context is used to reduce the number of rules related to this object which are searched from the rule base.

5 KEshell++: A Knowledge Engineering Shell with a Seamless Integration of Rules and Objects

5.1 Knowledge objects

When heuristic rules are embedded within an object, the object can infer on these rules to provide heuristic answers when receiving queries from other objects. Such an object is called a *knowledge object.*

A knowledge object consists of at least three parts: data items, inheritance hierarchy, and rules. Methods can be implemented in forms of rules, or as a fourth component. Both rules and methods can be specified as public to allow global access or as private to prevent external visiting and modification.

Knowledge objects seem to fall into the category of incorporating rules into objects. However, we argue that a seamless integration should also provide facilities to deal with objects embedded within rules, and therefore display the behaviour of intelligent objects as defined in Section 4. The KEshell++ architecture designed in the rest of the section will demonstrate such a seamless integration.

KEshell++ is a programming environment under development. Our general research plan is to design a programming language based on C++ which will permit seamless integration of object oriented design and rule-based reasoning, and develop knowledge acquisition capabilities which will automatically generate a meta knowledge base[2] from the source code. The project is built on the authors' previous work dealing with knowledge representation and acquisition for expert systems [Wu 91, Wu 92] and object-oriented software engineering environments [Kraemer & Schmidt 89, Kraemer & Schmidt 90, Schmidt 91, Karagiannis *et al.* 93, Schmidt & Zimmermann 94]. Section 5.2 outlines our existing work and Section 5.3 describes how we are extending it in the on-going project.

5.2 Rule schema + rule body and SIKT

5.2.1 Rule schema + rule body

Rule schema + rule body [Wu 94a] is an alternative representation language to rule-based production systems based on an integration of rule-based and numerical computations. Rule schemata in the language are used to describe the

[2]To avoid confusion between the terms knowledge base for information describing the internal description of the components in a system and their architectural framework, and the knowledge bases for expert systems, we use the term meta knowledge base for the former.

hierarchy among nodes or factors in domain reasoning networks. The computing and inference rules are comprised in the rule bodies, which are used to express specific evaluation methods for the factors themselves and for their certainty factors. A factor in rule schema + rule body can be a logical predicate or a variable whose value is either discrete (set-valued) or continuous (numerical).

In each rule body, there may be one or more inference rules similar to those in production systems. These rules may include instructions for numerical computation or an uncertainty calculus. All the rules in a rule body are used to determine the value of the conclusion factor in its corresponding rule schema and/or the certainty factor (CF) of these conclusions. When the concluon factor is a logical assertion, the rule body can be used to compute the CF of this assertion. When the conclusion factor is a variable, the rule body is used both to evaluate the variable's value and its CF. Thus, the CF computation can be processed in the same way as the evaluation of non-logical factors, both being explicitly expressed in rule bodies. When all the factors in a domain expertise are logical assertions and all the rule bodies have the same rules for computing CFs, the inexact inference then behaves similarly to the normal implementation approach in existing expert systems. When all the factors are numerical variables and no uncertainty calculus is needed, all the rule bodies will be used to express computation models and a rule schema plus its rule body is analogous to a procedure or function in conventional programming. Therefore, a knowledge base in this context, which is composed of a number of procedures and functions for numerical computation, plus the inference engine which solves user problems by using the knowledge base and is analogous to a main procedure, can have the same function as a conventional algorithm-based program. This feature of the rule schema + rule body language supports a feasible way to integrate software engineering with artificial intelligence.

A knowledge engineering shell, *KEshell* [Wu 91, Wu 92], with a powerful inference engine [Wu 93] has been designed to support this language.

A rule body may contain hundreds of computation and inference rules and is associated with a rule schema. Rule schemata in the rule schema + rule body language, which correspond to a domain reasoning network of the hierarchy among all the factors involved in a knowledge base, provide useful information about the structure of a (possibly very large) knowledge base, and therefore is an important source of information for the meta knowledge base in KEshell++.

5.2.2 SIKT: A structured interactive knowledge transfer program

SIKT [Wu 95] is a *Structured Interactive Knowledge Transfer* program designed and implemented in KEshell. It can automatically build executable knowledge bases out of direct dialogue with domain experts. As the dialogue process is structurally engineered, a domain expert does not need to know much about knowledge engineering or programming languages. All the expert needs to do is answer the questions asked by SIKT. SIKT builds up a factor dictionary and a reasoning network to describe the logical relationships among the factors. The expert can specify both numerical computation and logical inference during the dialogue.

SIKT first acquires factors and their logical relationships and then does consistency checking and rule body acquisition. Factors are put in a factor dictionary, and their logical relationships are described in forms of rule schemata. A knowledge base acquired this way can be divided into two parts: a meta component comprising the factor dictionary and rule schemata, and a rule bodies component for actual computation and inference during problem solving. The factor dictionary can contain various types of information about the factors, such as their value domain and constraints, and is thus also a useful source of information for the meta knowledge base.

5.3 Integration of objects and rules and automatic generation of useful information

KEshell++ is based on the rule schema + rule body language and the SIKT program. We are extending its capabilities in the following ways:

- Incorporate classes into rule schema + rule body as factors. The factor dictionary set up by SIKT

will contain classes as well as the original, simple factors.

- Design an independent object processing module in C++, which will implement methods of classes in the form of facts and rules.

  Message passing in this module can be treated as chaining on these rules, and therefore the existing inference engine in KEshell can be called. In the meanwhile, object construction and inheritance processing in the rules in rule schema + rule body will be passed to the object processing module for handling.

- Extend the UNIX Emacs editor as a frontend for our system.

  Existing editing support typically already comprises graphic template editing and browsing facilities for class signatures (their names and interface represented as a collection of acceptable messages, also a distinction between public interface and private functions). The extension will support the acquisition of methods of classes in the form of facts and rules.

- Compile an extendible library of algorithm fragments [Rich & Waters 90, Spinellis 93] and classes implemented with the above editor and corresponding documentation for the programmers to refer to and use.

  This will enable the programmers to build more powerful programs to solve more complex problems by reusing existing components in the library.

- Design a generation engine to produce a meta knowledge base from the source code edited with the above editor and the information acquired via SIKT.

  As discussed in Section 5.2, the factor dictionary and rule schemata will be the main part of the meta knowledge base. If the improved SIKT program in this project is used to build a knowledge base in an interactive way, the dictionary and rule schemata will have been generated by SIKT. However, if SIKT is not invoked during program construction and the programmer prefers to adopt a common editor or the specific editor above to edit their programs, rule schemata and the factor dictionary will need to be generated by the generator. Some work has already been done along the direction of generating rule schemata from concrete rules [Sutcliffe & Wu 94]. The generator will collect all the factors involved in the rule schemata, and produce an editable dictionary framework for the domain professionals or programmers to provide information about each factor's constraints. Whether a factor has been defined in the rule schemata and its value type will be inferred from the concrete rules.

- Provide an intelligent retrieval and reasoning engine for the programmers and end users to browse the meta knowledge base and make queries.

  This engine will answer questions related to the factors defined in a dictionary, and the structure of a knowledge base in terms of rule schemata, and concrete rules associated with each schema.

KEshell++ is being implemented in C++, and will be tested on some realistic application domains including large-scale telecommunication networks [Bapat 94].

6 Conclusions

Rule-based programming is the dominant programming paradigm in AI research and applications. Since its insufficient engineering power in structural representation and for large systems, we have discussed its integration with object technology, a powerful technology from software engineering and the database community. A new type of object, knowledge objects, is defined along with intelligent objects, and an extended object model and an on-going project, KEshell++, to implement such knowledge objects have also been outlined. When a knowledge base gets larger and larger, reuse of existing modules and encapsulation of physical objects become more and more important in the design and maintenance of the knowledge base. True improvements in large AI systems need engineering devices, and intergration of objects to provide efficient engineering facilities is clearly an important direction to take.

Acknowledgements

The authors are indebted to the two anonymous reviewers and in particular the editor Matjaz

Gams for their constructive comments on the paper.

References

[Atkinson *et al.* 92] Atkinson, Bancilhon, De Witt, Dittrich, Maier, and Zdonok, The Object-Oriented Database System Manifesto, *Building an Object-Oriented Database System: The Story of O2*, F. Bancilhon, C. Delobel, and P. Kanellakis, Eds., Morgan Kaufmann Publishers, 1992, 3–20.

[Bapat 94] S. Bapat, *Object-Oriented Networks: Models for Architecture, Operations, and Management*, Chapter 23, Prentice Hall, 1994.

[Barbier 92] F. Barbier, Object-Oriented Analysis of Systems through Their Dynamic Aspects, *Journal of Object-Oriented Programming*, **5**(1992), 2: 45–51.

[VanBiema 90] M. vanBiema, The Constraint-Based Paradigm: The Integration of Object-Oriented and the Rule-Based Paradigms, *PhD Thesis*, Columbia University, USA, 1990.

[Chandrasekaran 92] B. Chandrasekaran, Task-Structure Analysis for Knowledge Modelling, *Communications of the ACM*, **35**(1992), 9: 124–137.

[Cattell *et al.* 91] R.G.G. Cattell *et al.*, Next Generation Database Systems, *Communications of the ACM* (special section), Vol. 34, No. 10, 1991.

[Coad & Yourdon 91] P. Coad and E. Yourdon, *Object-Oriented Design*, Yourdon Press, 1991.

[Dillon 93] T. Dillon and P.L. Tan, *Object-Oriented Conceptual Modelling*, Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[Eckel 93] B. Eckel, *C++ Inside & Out*, McGraw-Hill, 1993.

[Giarratano 93] Joseph Giarratano, *CLIPS User's Guide* (CLIPS Version 6.0), Lyndon B. Johnson Space Center, Information Systems Directorate, Software Technology Branch, NASA, USA, 1993.

[Graham 93] I. Graham, Migration Using SOMA: A Semantically Rich Method pf Object-Oriented Analysis, *Journal of Object-Oriented Programming*, **5**(1993), 9: 31–42.

[Hedin 89] Hedin, An Object-Oriented Notation for Attribute Grammars, *ECOOP*, 1989, 329–345.

[Henderson-Sellers 92] B. Henderson-Sellers, Object-Oriented Information Systems: An Introductory Tutorial, *The Australian Computer Journal*, **24**(1992), 1: 12–24.

[Hoare 89] C.A.R. Hoare, An Axiomatic Basis for Computer Programming, *CACM*, **12**(1989), 10: 576–583.

[Jacobson 92] Jacobson, Christerson, Jonsson and Overgaard, *Object-Oriented Software Engineering*, Addison-Wesley, Readings, Mass., 1992.

[Karagiannis *et al.* 93] Dimitris Karagiannis, Franz Kurfess, and H.W. Schmidt. *The Next Generation of Information Systems: From Data to Knowledge*, Volume 611 of *Lecture Notes in Artificial Intelligence*, chapter Knowledge Selection in Large Knowledge Bases: 291–310. Springer, 1993. Selected from IJCAI 1991.

[Kraemer & Schmidt 89] B. Krämer and H.W. Schmidt, Developing Integrated Environments with ASDL, *IEEE Software*: 98–107, **1**(1989).

[Kraemer & Schmidt 90] B. Krämer and H.W. Schmidt, Architecture and Functionality of a Specification Environment for Distributed Software. In *IEEE Fourth International COMPSAC*: 617–622, Chicago, 1990.

[Kwok & Norrie 94] A.D. Kwok and D.H. Norrie, Integrating a Rule-Based Object System with the Smalltalk Environment, *Journal of Object-Oriented Programming*, **6**(1994), 9: 48–55.

[Medeiros 91] C.B. Medeiros and P. Pfeffer, Object integrity using rules, *ECOOP*, 1991, 219–230

[Meyer 89] B. Meyer, *Object-Oriented Software Construction*, Prentice Hall, 1989.

[Meyer & Nerson 90] B. Meyer and J-M Nerson, *Eiffel: The Libraries, TR-EI-7/L1*, Interactive Software Engineering Inc., Version 2.3, 1990.

[Meyer 92] B. Meyer, *Advances in Software Engineering, Object-Oriented Series*, Prentice-Hall, Englewood Cliffs, New Jersey, 1992, 1–50.

[Minsky 85] M. Minsky, A Framework for Representing Knowledge, *Readings in Knowledge Representation*, R.J. Brachman and H.J. Levesque (Eds.), Morgan Kaufmann, 1985.

[Moss 94] C. Moss, *Prolog++: The Power of Object-Oriented and Logic Programming*, Addison-Wesley, 1994.

[Odell 93] J. Odell, Specifying Requirements Using Rules. *Journal of Object-Oriented Programming*, **6**(1993), 2: 20–24.

[Ramakrishnan 93] S. Ramakrishnan, An Object-Oriented Design for Resource Allocation Problems, *Masters Thesis*, Monash University, Sept. 1993.

[Ramakrishnan 94a] S. Ramakrishnan, Two Layers of Constraints for an Extended Object Model in Eiffel, *Proceedings of the Technology of Object-Oriented Languages and Systems (TOOLS) USA*, Santa Barbara, Aug 1994, 115–124.

[Ramakrishnan 94b] S. Ramakrishnan, An Object-Oriented Design for Modelling Business Rules in Resource Allocation Jobs, *Proceedings of the Object-Oriented Information Systems (OOIS 94)*, London, Dec 1994, 105–113.

[Ramakrishnan 94c] S. Ramakrishnan, Quality Factors for Resource Allocation Problems - Linking Domain Analysis and Object-Oriented Software Engineering, *Proceedings of the First International Conference on Software Testing, Reliability and Quality Assurance (STRQA 94)*, Delhi, Dec 1994, 68–72.

[Rich & Waters 90] C. Rich and R.C. Waters, *The Programmer's Apprentice*, Addison Wesley and ACM Press, USA, 1990.

[Rumbaugh *et al.* 91] Rumbaugh, Blaha, Premerlani, Eddy and Lorenson, *Object-Oriented Modelling and Design*, Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

[Schmidt 91] H.W. Schmidt, Prototyping and Analysis of Non-Sequential Systems Using Predicate-Event Nets, *Journal of Systems and Software*, **15**(1991), 1: 43–62

[Schmidt & Zimmermann 94] H. Schmidt, with W. Zimmermann, A Complexity Calculus for Object-Oriented Programming, *Journal of Object-Oriented Systems*, **1**(1994).

[Spinellis 93] D. Spinellis, Implementing Haskell: Language Implementation as a Tool Building Exercise, *Structured Programming*, **14**(1993), 1: 37–48.

[Sutcliffe & Wu 94] G. Sutcliffe and X. Wu, Extracting Rule Schemas from Rules for an Intelligent Learning Database System, *Proceedings of the Seventh Australian Joint Conference on Artificial Intelligence (AI'94)*, C. Zhang and J. Debenham (Eds), World Scientific Publishers, 1994, 132–139.

[Wong 90] Limsoon Wong, Inference Rules in Object Oriented Programming Systems, *Deductive and Object-Oriented Databases*, W. Kim, J.-M. Nicolas, and S. Nishio (Eds.), Elsevier Science Publishers B. V., North-Holland, 1990.

[Wu 91] X. Wu, *KEshell*: A "Rule Skeleton + Rule Body" Based Knowledge Engineering Shell, *Applications of Artificial Intelligence IX*, M.M. Trivedi (Ed.), SPIE Press, U.S.A., 1991, 632–639.

[Wu 92] X. Wu, *KEshell2*: An Intelligent Learning Data Base System, *Research and Development in Expert Systems IX*, M.A. Bramer and R.W. Milne (Eds.), Cambridge University Press, U.K., 1992, 253–272.

[Wu 93] X. Wu, *LFA*: A Linear Forward-chaining Algorithm for AI Production Systems, *Expert Systems: The International Journal of Knowledge Engineering*, **10**(1993), 4: 237–242.

[Wu 94a] X. Wu, *Rule Schema + Rule Body*: A 2-Level Representation Language, *International Journal of Computers and Their Applications*, August 1994, 49–59.

[Wu 94b] X. Wu, Developing an AI Curriculum for Computer Science Majors, *AXIS: The UCISA Journal of Academic Computing and Information Systems*, **1**(1994), 3: 18-23.

[Wu 95] X. Wu, SIKT: A Structured Interactive Knowledge Transfer Program, *Proceedings of the 8th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems*, Melbourne, June 5-9, 1995.

[Zhao 94] Liping Zhao, ROO: Rules and Object-Orientation, *TOOLs Pacific '94 Technology of Object-Oriented Languages and Systems*, 1994, 31–44.

Modeling Affect: The Next Step in Intelligent Computer Evolution

Steven Walczak
University of South Florida
4202 E. Fowler Ave., CIS 1040, Tampa FL 33620
Phone: 813 974 6768
E-mail: walczak@bsn.usf.edu

Keywords: affect, emotion, machine learning, adaptation, problem solving

Edited by: Marcin Paprzycki
Received: May 8, 1995 **Revised:** November 16, 1995 **Accepted:** November 30, 1995

Artificial intelligence has succeeded in emulating the expertise of humans in narrowly defined domains and in simulating the training of neural systems. Although "intelligent" by a more limited definition of Turing's test, these systems are not capable of surviving in complex dynamic environments. Animals and humans alike learn to survive through their perception of pain and pleasure. Intelligent systems can model the affective processes of humans to learn to automatically adapt to their environment, allowing them to perform and survive in unknown and potentially hostile environments. A model of affective learning and reasoning has been implemented in the program FEEL. Two simulations demonstrating FEEL's use of the affect model are performed to demonstrate the benefits of affect–based reasoning.

1 Survival of Intelligent Systems

The field of artificial intelligence (AI) has made great advances the past decade, but there is still a debate over the use of the word "intelligent" to describe the systems produced from AI research (Searle 1980 & 1990). In contrast to Searle's negative view of the quality of intelligence in AI systems, both expert systems (Hayes–Roth & Jacobstein 1994) and neural networks (Widrow et al. 1994) are being broadly applied in scientific, engineering, and business domains to take advantage of increased quantity and quality of knowledge in decision making processes. With expert systems and other AI technologies being accepted and applied world–wide, what will AI research try to produce next? One of the long standing goals of hard AI is to produce an autonomous intelligent system, that is, a robot or some other artifact which can learn and adapt to its environment while performing other functions which require intelligent cognitive ability. Expert systems have succeeded in emulating human experts functioning within very narrowly defined domains, but how can intelligent systems function effectively in a dynamically changing environment?

AI research has followed a path of reverse evolution as shown in Figure 1. Problem solving tasks which require years of education and experience for humans have proven to be solvable by AI–oriented machines. Tasks which human beings take for granted such as seeing (image understanding) and talking, have proven to be extremely difficult problems to solve using machines. As these obstacles (perception and communication) to producing an autonomous intelligent system are overcome, another more difficult obstacle looms.

Autonomous intelligent systems such as tactical planners, robots, and autonomous vehicles need to be able to adapt to unknown situations which will arise in their domain environments. AI systems operating in the real world must learn to distinguish between beneficial and harmful objects if they are to survive. Current technology for autonomous systems focuses on the attainment of a specific goal (Arkin 1995, Chatila & Giralt 1987, Findler & Ihrig 1987) (e.g., moving from point A in the environment to point B) and assumes that all domain hazards are already known by the

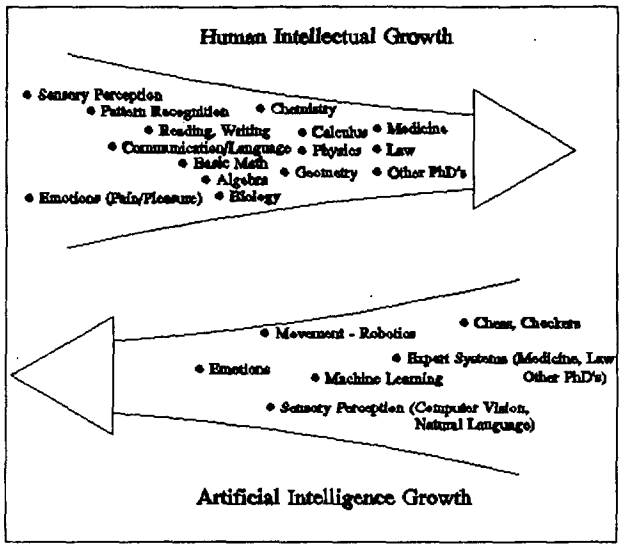


Figure 1: Evolution of human and machine intelligence

autonomous system's knowledge base. Some recent research in autonomous vehicles and robotics uses the term behavior to identify and describe the controlling programs of the robots, but these behaviors are still focused on enabling the vehicle or robot to maneuver in its environment and manipulate physical objects (Montgomery et al. 1995). Since the world is constantly changing, an autonomous system programmer would have to be omniscient to identify every possible situation and object that will be encountered by the autonomous system.

The solution to this problem is to make the autonomous tactical planners and robots capable of adapting to their environment. Adaptability is achieved through learning, but current learning techniques are too high level. Machine learning programs (e.g., AM and BACON (Michalski et al. 1983) and other more recent programs (Langley & Simon 1995)) have already demonstrated the acquisition of expertise in complex cognitive tasks, but machine learning research must address the more fundamental issue of species survival.

Why doesn't a human child grab a pot of boiling water with his hands more than once? Through personal experience or by second–hand learning, the child knows that the pot causes pain. This ability of humans to learn how to avoid pain and seek pleasure, learning through affect (emotions), provides an excellent model for allowing computers to learn to survive in an unknown environment by avoiding harmful situations and seeking beneficial situations. Plutchik (in Livesey 1986) and others (Frijda 1986, Lang 1983 & 1987) concur that emotions serve an adaptive role in helping organisms deal with key survival issues posed by the environment.

Reasoning with an affect–based model may produce additional benefits besides enabling intelligent systems to adapt to their environments. Finkel (1995) claims that higher level reasoning in humans can be a direct consequence of affect. Specifically, jurors and judges must interpret data to determine the proper application of law. While this may seem a purely objective task, Finkel provides several demonstrations of the influence of affect in jurors' perception of events and in jurors' evaluation of when laws should be applied or nullified.

Several artificial intelligence researchers have indicated their belief that autonomous computer systems need to utilize affect. In a personal comment, John Nagle states, "Robots that operate in the real world need mechanisms that implement fear and pain to survive." Sloman and Croucher (1981) claim that the need to cope with a changing and unpredictable world implies that computers will have emotions. The implementation of a learning system modeled on human learning via affect allows an intelligent computer system to adapt and survive in a previously unknown and currently changing environment. The affect learning system permits an intelligent computer system to differentiate between harmful and beneficial objects in its environment. This paper presents the development of a methodology for modeling affect based learning in computer systems. The method has been implemented in the program FEEL, Fuzzy Environment Emotion Learning. Results from this implementation are presented.

2 Previous Computer Models of Affect

During the 1960's and early 1970's, several emotion simulator programs, including ALDOUS (Loehlin 1968), came into existence. These programs were used to predict emotional outcomes for a given stimulus in a particular emotional setting. Two major deficits of these programs are the finite modeling method used and the passive nature

of the programs. Each of the emotion simulators represented emotions as a set of discrete values. Frijda and Swagerman (1987) and others (Sloman & Croucher 1981) indicate the uncertain nature of a real world environment. Because of this uncertain nature, the use of discrete values by previous emotion simulators yields an unrealistic approximation of the real world.

Pribam (in Livesey 1986) states that emotions precipitate action. Previous emotion models have been used to merely judge the change to an emotional state produced by a particular event. Intelligent systems will require guidance through interpretation of events in the environment and subsequent suggestions for appropriate actions. The actions must then be performed by actuators controlled by the system.

Early research which simulated the affect related effects of hunger was conducted by Grey Walter (see Asimov & Frenkel 1985), whose mechanical "tortoises" would find an electrical outlet when their batteries ran low, to replenish their diminished energy supply. The anthropomorphism of stating that the tortoise was hungry and that it was feeding itself has been applied many times. Normal behavior patterns which are operative during times of adequate system energy are altered to relieve the shortage condition. An analogous situation exists for diabetic humans who must immediately seek out a food source when their blood sugar levels are too low. Walter's tortoises were an early demonstration of the benefits of affect–based reasoning, but were limited to dealing with the single affect related issue of "hunger". Intelligent systems require this same type of reasoning capability, but with a broader overall application to enable the systems to deal robustly with dynamic environments.

3 Psychological Background for Affect Modeling

Research in psychology and sociology on the topic of affect and how affect effects decision making is extensive and a complete summary is beyond the scope of this paper. Research which directly supports or has motivated the developed model of affect and affective behavior is presented.

- **Genetic components of affect.**

  - The *visual cliff* experiments performed by Gibson and Walk circa 1960, were originally used to demonstrate the perception abilities of infants. A corollary proposition of these finding with regard to affect is that human beings have some innate or built in affect mechanisms for avoidance of pain (and death).

  - A social psychology experiment by Schwartz (1992) has demonstrated that affect– based perception and attitudes are largely universal in nature. Different social cultures have very similar affective values.

- **Production and modification of affect.**

  - *Integration Theory* (Anderson, 1991) states that external stimuli are processed with a valuation function, which could be affect–based or have affect as a component, to represent the stimulus internally and then transformed via an integration function into an external response. Integration theory is supported by the *Reinforcement Theory of Emotions* (Simonov 1986) which proposes emotions as a representative measure of how external objects can enable a person to satisfy (or attain) goals.

  - Anderson (1991) claims that man's motivational systems are based at least in part upon affective processes and that these processes should account for both *primacy* and *recency* effects. The primacy principal implies that attitudes are based most strongly on the first several interactions a person has with a particular object. Recency implies that basal attitudes must be adaptable to account for changes in the environment.

  - *Information Theory* (in Hunter et al. 1984) claims that the magnitude of change in emotion beliefs is proportional to the difference between the current affective state and any received data. Any change to the affective state is in the direction of the received data. This claim by Information Theory is also supported by the Reinforcement Theory of Emotions.

  - Another concept proposed by Information Theory (in Hunter et al. 1984) is that the effect of input from the environment accumulates over time, making changes to existing attitudes

more difficult as additional encounters with similar objects or situations are encountered.

- **Other factors.**

- Livesey (1986)states that there is an intransigence of problems associated with the study of emotion. Among these problems is the task of defining which emotions should be considered as primitive emotions. Lang (1987) has indicated that two primitive emotions are sufficient for modeling observable interactions between a person and the environment.

- *Social Psychology* experiments have demonstrated that "neutral moods" have greater susceptibility to change than more positive or negative moods (Schwarz et al. 1991).

- Frijda (1986) and others (Forgas 1992, Minsky 1994) have claimed that emotions have a definite information content and are used in cognitive reasoning. As an extension to this claim, Hommers and Anderson (1991) demonstrate experimentally that some moral rules have algebraic forms.

The listed items above form part of the foundation for the cognitive model of affect developed in the research described in the next sections. Viewed as a whole, the background items imply that affect has some rules or methodology which can be used for determining the affective response of a person or system to its environment.

4 A Cognitive Model of Affect

The Reinforcement Theory of Emotions discussed in Section 3 proposes that emotions are a probabilistic measure of the effect of external stimuli towards goal satisfaction. Identifying objects in the environment (through affective values) which can assist a system in satisfying its goals is precisely what the computer must learn to be able to operate efficiently and to survive in a changing environment.

Several critical factors which are present in the human affect based learning mechanism must be accounted for by any model of affect to be used for computer learning. These factors are: which specific types of affect (particular emotions) need to be modeled to allow the computer the greatest chance for survival, what should be the initial attitude of the system corresponding to the way a human reacts to an unknown object, how do affect based perceptions change over time, and how to account for the effect of situational relevance. Each of these factors is discussed below.

4.1 Architecture and cognitive foundations

Prior to resolving each of the issues above, the general architecture of an affect–based learning system is examined. The affect value of any domain situation is composed of the sum of the current affective state of the autonomous system, the previous affect value of the current object, and the current affect value of the action which is occurring. Both objects and actions have affect values. The affect value of an object refers to the system's perception of an object from past learning experiences. Action affect values are used to alter the affect value of objects when specific situational actions occur.

A diagram of the system architecture is shown in Figure 2. The FEEL program interprets input from the environment and determines the system's affective state by using a heuristic rule–based production engine to determine the effect an action and an object, each with affect values, to the current affective state of the system. Details of the system will become apparent as the traits of the human emotion system are discussed below.

Simonov (1986) emphasizes the close connection between affect and the needs of an organism or, in our case, a computer. FEEL uses a planar coordinate system for binding affect values to objects which are encountered in the environment to simulate Simonov's connection between affect. Planar coordinates were chosen to model the binary representation of primitive emotions suggested by Lang (see Section 3). The first axis of FEEL's planar coordinate system represents the relative friendliness of an object and can be thought of as a love/hate aspect. The second axis represents the usefulness of an object to the system in solving problems which the system encounters and can be thought of as a desire/fear aspect. These two affect scales were chosen to be the primitives of the system because they most

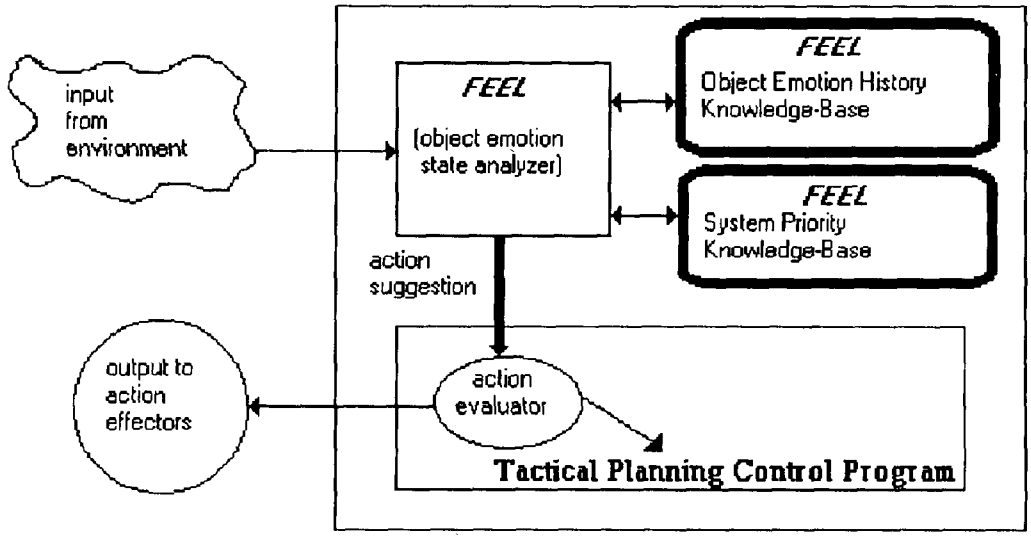


Figure 2: FEEL system architecture

accurately reflect the affective states required to learn to survive and thus encompass the *needs* of the computer system. As an example, an object (such as a user of the computer system) which has previously satisfied a non–life–sustaining priority of the computer would have a positive friendliness value and a slightly positive usefulness value, since the object has helped the computer to realize one of its priorities, but has not had an effect on the overall survival of the computer system.

Continuous values are used for each pair of coordinates in the planar coordinate model of affect. The two values assigned to each object represent the learned affective state of the computer towards the object. These values can be used in conjunction with each other or separately as the heuristics of the situation demand. This is in contrast to the more deterministic approach used in the earlier emotion simulators where the largest discrete valued emotion controls all reactions.

When a computer system using the FEEL affect–based learning method first encounters an object in its environment, what should be the initial affect value assigned to that object? If the newly identified object can be associated with another object which has already acquired an affect value, then the affect value of the previously learned object is transferred to the new object. For example, the computer identifies a human labeled 'User B' as an unknown object and 'User B' is carrying a lighted torch so that he can see. The computer has previously identified 'fire' as having a strong negative affect value since a fire nearly destroyed the computer in a past encounter. 'User B' would acquire the negative affect rating of 'fire' as an initial affect value. If no association to previously identified objects can be made, then the base affect value used by FEEL is strictly neutral which is then modified by consequent actions.

By starting every new object from neutral, each object will be able to control the affect value assigned to itself without preconceived prejudices. A more defensive posture can be assumed by starting all new objects with slightly negative values for each of the planar coordinates. Initializing the affect learning system with negative values would cause FEEL to pursue an evasive course of action upon encountering an unknown object in its environment. Conversely, starting objects with a positive value would model the behavioral ideals of trust and friendship.

A key component of the affect–based learning paradigm is the model of attitude change. The way in which the affective state associated with an object changes with experience is what permits the computer system to correctly adapt to its environment. The model of attitude change is based upon the Information Theory concept that emotions change proportionally to the difference between the current affective state and the received data. Additionally, any change to the current affective state is in the same direction as the affec-

tive value of the received data (perceived action).

The FEEL model of attitude change uses another idea from Information Theory, that the effect of input from the environment accumulates over time, thus making well founded attitudes harder to change. A well founded attitude is an affective state associated with an object which has been supported by a large number of prior interactions with that object.

Hence, the model of attitude change interprets externally observable data. This data consists of observable actions which are performed by an object in the environment. Each action has affect values attached to it which depend on the degree of benefit or harm that the action produces for the computer system. The affect value of the action is then used to modify the current affective state associated with the observed object in the direction, positive or negative, of the affect value of the action that is observed. A record of the number of interactions which have occurred with each object is maintained and used to scale the degree of change in the affective state (see equation (3) in Section 4.4). The FEEL affect learning system is capable of dynamically assigning an affect value to actions which have not been previously encountered by evaluating the effect of the action with regard to the current priorities of the FEEL system.

Each object record is stored in a knowledge base of objects which have been encountered. This knowledge base is updated after every event the FEEL system observes/experiences. Figure 3 displays in a graphical format the model of attitude change which has been developed. The x–axis represents the strength and value of the current affective state and the y–axis represents the degree and direction of change in affective state. The heavy line in the graph represents conflicting input values (e.g., an object which has a positive affect value performs an action which has a negative affect value) and the light line represents supporting or similar input values. As can be seen from Figure 3, newly learned affect values, those closest to the y–axis, are subject to the greatest degree of change while more well founded affect values, farther from the y–axis, only change by a small amount. This model captures another peculiarity of human affect–based learning. Humans react more strongly to conflicting input than to supporting input. This means that a positive affect value associated with an object becomes only slightly more positive when a positively valued action is produced by the object, but a negatively valued action produced by the object causes a more significant change in the negative direction.

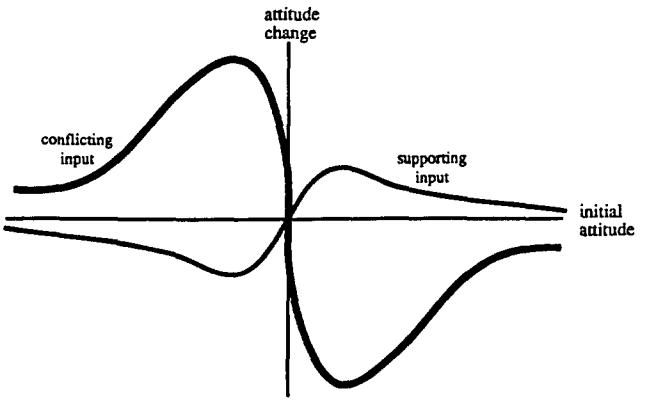


Figure 3: Model of attitude change

4.2 Psychological survey of attitude change

A psychological survey to evaluate the FEEL model of attitude change was conducted at a large state university in the United States on a sample of 88 students enrolled in introductory psychology classes. The survey presented eighteen different situations and the students were asked to mark a bar on a sliding five point scale to indicate the type of response each student would have to the situation (from strongly negative to strongly positive). Although the survey response forms only had five anchor points, the use of the sliding scale enabled the research to interpret the data as a nine point scale. Additional demographic data and a self evaluation of social behavior (introverted versus extroverted) was also acquired. The survey was repeated at a small liberal arts university on a smaller sample, producing similar results.

The results of the survey matched the model (see Figure 3) identically to the right of the y–axis, however, the human subjects tended to resist changing a negative attitude more strongly than a positive attitude when faced with input

which supported a contradictory affective state. This defensive posture is more realistic for survival. If someone has been threatening to kill you for several months and suddenly sends you a box of your favorite chocolates, it would be wise to proceed with caution before consuming the sweets. To model this affective trait of humans, the unwillingness to change negative affect related attitudes, FEEL uses an unbalanced weighting mechanism. Larger weights (0.085) are attributed to negative affective states which indicate a potentially harmful object and smaller weights (0.045) are attributed to positive affective states which indicate beneficial objects. The weights are then used to scale the degree of change in the affective state. This weighting and scaling causes changes to an affect value to be reduced for affect values associated with an object that has larger weights (i.e., the effect of changes to affect values of negatively valued objects is smaller than to other similarly experienced positive affective values).

4.3 Situational relevance

The last element to be included in the cognitive model of affect used by FEEL is the effect of situational relevance. Situational relevance refers to the fact that human affective behaviors are not just based on specific individual objects and actions, but are instead a conglomeration of established and recent affective states with the affective state of the current object. Situational relevance is analogous to the term "mood" used in social psychology (Forgas 1992) to describe enduring affective states which do not have a precise antecedent cause.

If the computer is having a good day with many encounters of positively rated objects and then receives some negative, but not survival threatening input from its environment, then the computer is more likely to continue in a good/positive, although somewhat reduced, affective state. This backward referencing is done in two ways. First, the current affective state of the computer is used in calculating the new affective state with respect to the current object. Second, the affect value of the previous action observed by the particular object is used to support or decrease the affect value of the current action. If an object has repeatedly performed good actions as judged by the computer, which results in a strong positive affective state for that object, and the object then performs an action which is perceived as being slightly harmful to the computer, then the computer "knows" that this single action is not normal for that object and scales down any affective state changes accordingly. If additional negative actions are then produced by the object, the computer identifies the new pattern of actions and begin changing the affective state associated with the object in the negative direction corresponding to the new set of actions. As an analogy, the situational relevance of an action may be viewed as a reverse simulated annealing process, where initial changes to well formed (established) weights are small, but as the affective state approaches neutral, more recent interactions with an object produce greater effects.

4.4 Heuristic equation of attitude change

The model of attitude change for affect values associated with objects can be summarized with equations (1-3), where O_i is the affective state associated with the object at time i, q is the quantity of previous interactions with this object, A_i is the affect value associated with the action just observed at time i, W_i is the unbalanced weight for the object at time i, S_i is the current affective state of the computer system at time i, and ε is an error term to account for minor situational variances. The $\pm$ symbol is used to indicate either a negative one for negative valued A_i or a positive one for positive A_i values. The constants (4, 5, and 10) were heuristically determined to maximize the affect learning rate.

$$E(O,i) = \frac{qO_i + \left[O_{i-1} + 4\left(\pm 1 - \frac{1}{A_i \pm 1}\right)\right]/5}{q+1} \quad (1)$$

$$Mood(i) = \frac{W_i S_i}{10} \quad (2)$$

$$O_{i+1} = E(O,i) + Mood(i) + \varepsilon \quad (3)$$

Equation (3) is used to calculate both the x (friendliness) and y (usefulness) affect values, thus the equation must be evaluated twice for each observed action in the environment.

It is possible to make small modifications to equation (3) without significantly altering the meaning of the attitude change model. For example, by using O_i^q instead of qO_i in equation (1) and subsequently normalizing the fractional value, produces the effect of making it more difficult to change the existing affective values associated with previously encountered objects. The apparent linear nature of equation (3) may cause concern due to the recent preference of averaging theory to additive models in psychology (Anderson 1991). The use of the O_{i-1} term in equation (1) enables the attitude change equation to effectively incorporate averaging. Finally, some research in psychophysics prefers to use power functions (Stevens power function) with variable weights to model changes in the degree of effect for positive and negative stimuli (Estes 1992). Again, a simple change in Equation (2) from $W_i S_i$ to $S_i^{W_i}$ provides the feel of power functions by modifying the scale without changing the intent of Equation (3).

5 Affect–Based Planning

The models of affect and attitude change described previously permit a computer system to identify and differentiate between beneficial and harmful objects in its environment. A methodology is required which allows a computer system to utilize the learned affect values of objects to adapt to its environment. The methodology used by FEEL is akin to Asimov's Rules of Robotics (Asimov 1950). The system maintains a knowledge base of priorities or system goals similar to the idea of motives introduced by Sloman and Croucher (1981). Each priority has a value associated with it so that the relative importance of the priorities can be determined. Sloman and Croucher state that motives are not static and the *needs* of a system can vary over time. Although those priorities which affect system survival are assigned static values, the other priority values change to reflect the changing needs/priorities of the computer system. Tests which can be used to evaluate whether a priority is being met are included in the knowledge base with each priority record. For example, a priority of maintaining high cpu utilization would include a test such as: if utilization is less than 85 then priority is not being satisfied.

The evaluation of priorities is performed by comparing observable data with a range of desired values recorded in the frame slot which holds the priority tests, such as the current and desired temperature of the computer or the current system state versus a desired goal state. The highest valued priority which is impacted by the current input data is activated. FEEL remembers the previously activated priority to aid in deciding whether to activate a new priority or to continue with the current priority. These priorities, while interacting with each other, use the affect values associated with objects to choose a course of action. Certain priorities, those which affect the survival of the system such as excessive temperature or electrical flux, can override the other priorities (these predefined critical priorities are similar to the genetically encoded pain/survival knowledge, such as the visual cliff described in Section 3). When a potentially dangerous situation is identified by the computer system, various courses of action to resolve the situation are attempted while still permitting the other priorities to execute actions. If harm to the computer system is imminent, then all other priorities are overridden, similar to the idea of proto–specialists proposed by Minsky (1986, 1994). This exclusive nature of certain priorities is required to prevent competition by other similarly valued priorities in a situation which is critical to the survival of the computer system.

The courses of action to resolve situations are chosen through two methods. Emergency and unknown situations are handled by a separate production system which contains a hierarchy of action sequences for resolving these types of situations. An example of an action sequence, for the situation where a rapid and unexplained increase in the environment temperature has occurred, would be to initiate contact with a human operator if one was available, activate any cooling systems under the computer's control (such as central AC), and finally if fire is suspected, activate the fire suppression system (automatic fire extinguisher). New action sequences can be learned by instruction from an external source. If an emergency exists and no action sequence is successful in resolving the situation, then the default action of seeking human help is initiated. Human helpers are chosen by the affect values associated with them

corresponding to their usefulness to the computer system.

When a situation arises that is within the scope of the tactical planner or robot using the affect–based learning system, then the second method of allowing the tactical planner to choose the appropriate action is used. If alternative action choices exists within the tactical planner, then FEEL can suggest which action is most beneficial to the computer system's goals.

6 Implementation and Evaluation of the FEEL Affect Model

Two simulations were performed to evaluate the effectiveness of FEEL's affect model in enabling AI systems to reason about complex dynamic environments. The first simulation performed concerns lower–level affective processes (i.e., any affective state which is directly correlated with system survival or pleasure). The second simulation, which has been performed more recently as part of the continuing research with affect–based modeling, concerns higher–level affective processes (i.e., using information content of affective states to assist or control cognitive decision making).

6.1 Intelligent computer operating system

The model of affect implemented in the FEEL program was tested by running the program in a simulation of an autonomous computer system environment. Instead of using a hypothetical tactical planner, the affect–based learning system was used to guide an intelligent computer operating system. The goal of this simulation was to evaluate the proposed affect model and the effectiveness of the model in allowing a computer to learn to adapt more appropriately to its external environment. Evaluation was performed by analyzing the effect of the FEEL affect–based action plans on system priorities versus a partial factorial study of the effect to system priorities from other action sequences.

Objects in this simulation were the users of the computer system as well as inanimate objects such as fire, water, heat, and cold. The priorities of the system followed typical operating system goals such as maximizing computer utilization and performance, as well as maintaining certain system requirements like temperature and flow of electricity. Object and action pairs were input into the system. An example of an object–action pair is: User A; login or User A; submit_job, as well as object–action pairs for inanimate objects such as: fire; hot. After each object–action pair was processed by the FEEL system, the affect values for the most recent object, action, and for the affective state of the computer system were displayed. When effectors were available, the actions specified in the simulation were carried out, including logging users into a Unix–based operating system and submitting job requests to the system.

Actions produced by the objects in the environment, the users of the computer, were evaluated according to their impact on the computer system's set of priorities. One of the priorities of this simulation was to maximize user turnaround. Upon detecting a decrease in CPU utilization due to an increase in the user population, FEEL requested new users to delay their use of the system. Users who ignored this suggestion were considered to be harmful to the computer system and consequently their requests of the system were given a lower priority than the more beneficial users.

Finally, a simulated rise in the computer room temperature was enacted. After finding no appropriate action resolving the situation, FEEL used its default mechanism to select specific users based on their affect ratings of usefulness to the computer system and requested their help.

If a specific user failed to aid the system, the affective state associated with that user was adjusted and a new user was selected from whom to request help. The FEEL program accurately evaluated the situations in the simulation and made appropriate adjustments, as represented in Figure 3, to the affective state of the corresponding objects.

6.2 Juror case evaluation

The second simulation of the FEEL affect model was performed in the domain of law. Objects in the environment were simulated jurors and actions were the evidence presented and statements made by the various attorneys. The action effector for this simulation was to cast a guilty or a

not guilty vote for the defendant. An example of an object–action pair for this simulation is Defendant; victim_mugging or Defendant; murder. Evaluation of the affect–based juror simulation was performed by using detailed case studies of trials (Finkel 1995). The simulation trials were all for violent crimes (e.g., murder) and used actual trials which had juror interviews to indicate the reasoning process of the human jurors. The outcome of the FEEL simulated jurors was compared against the decisions of human jurors. Jury deliberations are infused with affect (Finkel 1995) and their verdicts can produce additional affective responses (e.g., the Los Angeles riots following the original Rodney King trial).

In the previous simulation which used the affect model in a computer operating system environment, the affect states of the objects, system, and actions were able to directly define the affect state of the system and the system's corresponding course of action to maximize the current system objectives. The juror simulation was able to reasonably model jury outcomes when both simple and contradictory evidence was presented as affective objects. Simple evidence would be were there is an overwhelming quantity of evidence supporting a specific verdict. Simple evidence causes the *mood* or situational relevance trait of the affect model to move towards an affective state supporting the release or conviction of the defendant. The term contradictory evidence is used to indicate the presence of both positive and negative valued affective objects supporting the conviction of the defendant. For cases with contradictory evidence the affective state of the system fluctuates around a median value which averages the affect values of the crime with the presented evidence.

When compound evidence was presented, the affect model produced inconsistent affect–based verdicts, with the inconsistencies related to the current system priorities of each simulated juror. Compound evidence is when normal simple evidence is available for a conviction vote, but the affective situational relevance of the defendant's history is similar to a previous encounter defined for the system. Similar situational relevance means that the juror identifies or sympathizes with the defendant's situation. Lawyers are well aware that a sympathetic jury can change the objective outcome of a trial (e.g., this is why it took several months to determine the jury composition for the O. J. Simpson trial). More realistic results were obtained from the simulation by adding a third weight, -0.125, to the attitudinal change equation (2) which was used if the juror and defendant had a common experience. Additional research is needed to further evaluate the greater complexity of using affect for decision making when the cognitive decision is not directly related to survival (pain or pleasure) of the system.

7 Summary

In the computer simulation described in Section 6, which was repeated multiple times with different sequences of actions, FEEL demonstrated the capability of implementing a model of affect which allowed the computer to selectively adapt to its environment. While the number of action effectors was limited in the simulation, current machines can take advantage of the Internet or other wide–area networks to automatically place service calls to repair technicians or to automatically order replacement parts. Future research efforts will use FEEL in a robotic system which has a greater number of action effectors available for altering the system's environment in response to the affect values generated by FEEL.

Additional research is required to investigate the use of affective states which have been previously learned and their application in analogous situations. Analogy would permit FEEL to initialize the affect value associated with an object more appropriately, related to the context in which the object is encountered.

The models of affect and attitude change presented, enable intelligent computer systems to modify their reactions to objects and actions in their domains to produce a context–relevant response. When combined with the hierarchical priority structure described in the paper, the learning of appropriate affect values will increase the survivability of autonomous intelligent systems.

References

[1] Anderson, N. H. (1991) *Contributions to Information Integration Theory Volume I: Cogni-*

tion. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[2] Arkin, R. C. (1995) Intelligent Robotic Systems. *IEEE Expert*, 10(2), p. 6–8.

[3] Asimov, I. (1950) *I Robot.* Westminster, MD: Del Rey Books.

[4] Asimov, I. & Frenkel, K. A. (1985) *Robots, Machines in Man's Image.* New York: Harmony Books.

[5] Chatila, R. & Giralt, G. (1987) Task and Path Planning for Mobile Robots. In Wong & Pugh (eds.), *Machine Intelligence and Knowledge Engineering for Robotic Applications*, p. 299–330, Berlin: Springer–Verlag.

[6] Estes W. K. (1992) Mental Psychophysics of Categorization and Decision. In Geissler & Link & Townsend (eds.), *Cognition, Information Processing, and Psychophysics: Basic Issues*, p. 123–139, Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[7] Findler, N. & Ihrig, L. (1987) Analogical Reasoning by Intelligent Robots. In Wong & Pugh (eds.), *Machine Intelligence and Knowledge Engineering for Robotic Applications*, p. 269–282, Berlin: Springer–Verlag.

[8] Finkel N. J. (1995) *Commonsense Justice: Jurors' Notions of the Law.* Cambridge, MA: Harvard University Press.

[9] Frijda, N. (1986) *The Emotions.* Cambridge: Cambridge University Press.

[10] Frijda, N. & Swagerman, J. (1987) Can Computers Feel? Theory and Design of an Emotional System. *Cognition and Emotion*, Vol. 1, p. 235–258.

[11] Forgas, J. P. (1992) Affect in Social Judgments and Decisions: A Multiprocess Model. In Zanna (ed.), *Advances in Experimental Social Psychology Volume 25*, p. 227– 275, San Diego: Academic Press.

[12] Hayes–Roth, F. & Jacobstein, N. (1994) The State of Knowledge–Based Systems. *Communications of the ACM*, 37(3), p. 27–39.

[13] Hommers W. & Anderson N. H. (1991) Moral Algebra of Harm and Recompense. In Anderson (ed.), *Contributions to Information Integration Theory Volume II: Social*, p. 101–142, Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[14] Hunter, Danes, & Cohen (eds.). 1984. Mathematical Models of Attitude Change, New York: Academic Press.

[15] Lang, P. (1983) Cognition in Emotion: Concept and Action. In Izard, Kagan, & Zajonc (eds.), *Emotion, Cognition, and Behavior*, p. 192–226, Cambridge: Cambridge University Press.

[16] Lang, P. (1987) Fear and Anxiety: Cognition, Memory, and Behavior. In Magnusson & Ohman (eds.), *Psychopathology An Interactional Perspective*, p. 159–176, New York: Academic Press.

[17] Langley, P. & Simon, H. A. (1995) Applications of Machine Learning and Rule Induction. *Communications of the ACM*, 38(11), p. 55–64.

[18] Livesey, P. (ed.). (1986) *Learning and Emotion: A Biological Synthesis.* Hillsdale, NJ: Lawrence Erlbaum Associates.

[19] Loehlin, J. (1968) *Computer Models of Personality.* New York: Random House.

[20] Michalski, Carbonell, & Mitchell (eds.). (1983) *Machine Learning An Artificial Intelligence Approach.* San Mateo, CA: Morgan Kaufmann.

[21] Minsky M. (1986) *The Society of Mind.* New York: Simon and Schuster.

[22] Minsky M. (1994) Negative Expertise. *International Journal of Expert Systems Research and Applications*, 7(1), p. 13–18.

[23] Montgomery, J. F., Fagg, A. H., & Bekey, G. A. (1995) The USC AFV–I A Behavior–Based Entry in the 1994 International Aerial Robotics Competition. *IEEE Expert*, 10(2), p. 16–22.

[24] Schwartz, S. H. (1992) Universals in the Content and Structure of Values: Theoretical Advances and Empirical Tests in 20 Countries. In

Zanna (ed.), *Advances in Experimental Psychology Volume 25*, p. 1–65, San Diego: Academic Press.

[25] Schwarz, N., Bless, H., & Bohner, G. (1991) Mood and Persuasion: Affective States Influence The Processing of Persuasive Communications. In Zanna (ed.), *Advances in Experimental Psychology Volume 24*, p. 161–199, San Diego: Academic Press.

[26] Searle, J. R. (1980) Minds, brains, and programs. *Behavioral and Brain Sciences*, 3, p. 417–457.

[27] Searle, J. R. (1990) Is the Brain's Mind a Computer Program? *Scientific American*, 262(1), p. 26–31.

[28] Simonov, P. V. (1986) *The Emotional Brain.* New York: Plenum Press.

[29] Sloman, A. & Croucher, M. (1981) Why Robots Will Have Emotions. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, p. 197– 202.

[30] Widrow, B., Rumelhart, D. E. & Lehr, M. A. (1994) Neural Networks: Applications in Industry, Business and Science. *Communications of the ACM*, 37(3), p. 93–105.

The Extracellular Containment of Natural Intelligence: A New Direction for Strong AI

Richard L. Amoroso
The Noetic Institute, 120 Village Sq. #49,
Orinda, Ca, 94563-2502 USA
Phone: 510 893 0467
E-mail: ramoroso@hooked.net

Keywords: AI, conscious computing, molecular electronics, teleology

Edited by: Marcin Paprzycki
Received: May 16, 1995 **Revised:** November 27, 1995 **Accepted:** November 30, 1995

Attempts to mimic human intelligence through information processing alone have failed because human rationality contains an element of non-linear acausality - something left out of the design criterion of linear machine intelligence. Based on the fundamental premise that a noumenon of consciousness is an inherent teleology in the fabric of the physical universe; the architecture of a molecular quantum holonomic computer, can be designed to embody the physical elements of natural intelligence. Consciousness emerges within its core because the utility of the missing parameters of mind contained in the deeper ontology function as a carrier to simulate a platform of natural intelligence.

1 Introduction

Strong AI positions of "mind = computer" because of the general lack of progress, force opinions like the Dreyfus brothers comparing AI to a try at reaching the moon by climbing a mountain. Although there are obvious computational aspects of mind; computation is not sufficient for a conscious machine.

A subquantum ontology for a conscious computer, represents a radical new direction for AI research. This conceptualization of mind, also discards the similar "mind = brain" position in favor of a deeper teleological holism stemming from a radical reinterpretation of quantum field theory (Amoroso, 1996a, Amoroso & Martin, 1995). The standard Copenhagen model is an epistemological interpretation based on the phenomenology of measurment and thus limited by the uncertainty principle. Ontological interpretations seek an understanding deeper in the noumenon, or thing in itself, independent of perceptual reality.

Rather than one emergent identity, Quantum Brain Dynamics (QBD) (Ricciardi & Umezawa, 1967) in the nonlocal arena, suggests that mind is composed of three integrated base states: 1. Nonlocal elemental intelligence. 2. Cosmological ordering principle. and 3. QBD. For a detailed delineation see Amoroso, 1996a and Amoroso & Martin 1995. The physical nature of the Noetic Field sets aside the notion of an immaterial mind (Amoroso, 1996b, 1995b); providing for the extracellular containment of natural intelligence. The parameters of this universal mind matrix simulated within heterosoric stacks of charge transfer salts provide the correct molecular electronics to recapitulate the fundamental spacetime geometry of natural intelligence. When this core is made to resonate with laser interferometry at biological frequencies (Frohlich frequencies) (Frohlich, 1968, 1983, Amoroso, 1996c) the device operates as a conscious computer as natural intelligence emerges within it. The design of this conscious computer is a feasible new direction for strong AI.

2 Bose-Einstein Condensation

The paracrystaline nature of the vibration of electrically charged dipole protein molecules in the brain create highly ordered states. This represents a substrate for the vehicle of conscious awareness when in a coherent superposition coupled

to the cosmology of the noumenon. This ground state has been suggested to be a system of Bose-Einstein condensation (Frohlich, 1969, 1983, Marshall, 1989). Bose condensation allows the superposition of an infinite number of initial Pauli states into a coherent whole. Also called vacuum zero point fluctuations, ground states have been suggested as the mechanism of memory storage (Riccardi and Umezawa, 1967). The Bose condensate is a viable mechanism for the core of a conscious computer because it has the necessary degrees of freedom and provides a transition from linear/causal to nonlinear/acausal states.

Computation in the brain occurs in the holoscape manifold of dendritic microprocessing (Pribram, 1991). This structure is integrated with the top level of the Heisenberg spacetime raster where polarized molecules in neural Fermi states transduce sensory information into quasiparticles resulting in Bosonization. The local processing in the brain can be generally considered a system of Fermi interactions or particles obeying Fermi-Dirac statistics. Wave functions are said to be either symmetrical or antisymmetrical with the interchange of particle pairs. The Pauli exclusion principle states that Fermions due to intrinsic spin cannot occupy the same single-particle states as antisymmetrical spin half particles. In systems processing energy such as brain proteins, when a real Fermi particle like an electron moves through a dipole domain such as conformational translation along tubulin dimer molecules of a microtubule, it becomes clothed in a sea of virtual particles with a certain lifetime that it drags along with it. These complex particles are called quasiparticles (Bahm and Pethick, 1991). Under certain conditions quasiparticles containing an even number of Fermions can Bose condense. Bose condensation can produce superradiance and self induced transparency (Jibu, and Yasue, 1994). This transition is the top level of the triune nature of human intelligence. The linear causal nature of the entrainment of neurosensory events into the holoscape manifold must be transduced further into the nonlinear acausal domain. Any disordered thermodynamic process can be converted to holonomic coherence by this process.

Also in the phenomenology of Fermi QBD, the initial phases of holoscape entrainment into quasibosons has a mutual locality or same local chirality as induced by the quasiparticle production. Bosonization or Bose-Einstein condensation allows the process to go nonlocal and couple to the Noumenon state of elemental intelligence. The boundary conditions between states flips, utilizing spin or spinors. The Bosonization is reversible allowing for information to pass in each direction. In the lightcone gauge formulas the Fermi coordinates collapse to an n - 1 dimensional spinor field by a flipping of the boundary conditions of the original vector superfield. The equivalence of the two states is brought about by bosonization and refermionization in the correspondence in the change of boundary conditions in the world sheet. This conceptual explanation originates from superstring theory (Green et al, 1988).

Frohlich, 1968, 1983 describes coherence associated with a condensate not of material particles as in liquid Helium at cryogenic temperatures, rather of quanta of strongly excited collective polar modes of vibration in biological systems. The stabilization of this non equilibrium is achieved by coupling with an elastic field where excitation can be dampened and locked in. Such a sympathetic ordering via entrainment is well known in lasers, which also require a pumping mechanism to achieve coherence. Frohlich's original idea was that dynamical equilibrium represented by a limit cycle could be tuned by chemical/electrical stimulus and cause the collapse of the limit cycle. The triggered release of energy could then be harnessed to invoke large scale molecular events such as changes in the geography of QBD.

A precondition for consciousness in both the brain and a computer is the ordering and storing of information in the face of randomization. The challenge is to see if quantum systems self organize. Bose-Einstein condensates have the unique property of making coherent wholes by summing the behavior of many component parts which feedback on their elements and create a community. When cell membranes vibrate sufficiently to be drawn into the Bose- Einstein psychon matrix they are forming a coherent whole which resists degeneration by thermal chaos, which (ironically) gives rise to their movement in the first place. That is, something must supply the jiggling, and something must supply the ordering principle - one arises out of the other and then feeds back through the system. If electrical activity

of the neuron provides the energy to jiggle molecules which in turn emit photons; these photons synchronize jiggling and further photon emissions through superradiance (Dicke, 1956). This chain reaction is analogous to the pumping of a laser. The shift into the condensed phase depends on this molecular photon interaction. It is here where quantum wholeness radiates out over the entire structure. If a FQB Transition can be studied in single celled organisms it would signify that anything, including a computer, nearing the complexity of this biological system would be capable of conscious awareness. Such a quantal entity would however be limited in available states. The quantal state of mind postulated by Noetic Field Theory (Amoroso, 1996b) asks what the basic quantum of awareness is. The Einstein is a unit of measure signifying a mole of photons (Avogadro's Number). What magnitude of the unitary Einstein demarks the transition from awareness to self awareness? The boundary conditions of a condensing photon in a brain or computer system is enough. Rationality is not an issue at this level of monochromatic awareness and binary goals. Though critical mass arguments abound (a necessary condition for the decoherence and collapse of wave functions) all highly organized cells have a functional Holonom accessing the noumenon of consciousness inherent to the nature of their existence.

Coherent photon emission has been postulated to occur without a pumping mechanism by Dicke, 1956; and is called superradiance. The total Hamiltonian of this phenomena for biological systems has been described by (Jibu and Yasue, 1993) to have the collective dynamic properties required for this superradiance.

3 The Origin of Natural Intelligence

Three types of nonlocality may be defined: Spacial nonlocality and its complement temporal nonlocality, together which describe the entire phenomenological universe; and type III nonlocality, the timeless unity of spacetime. The principle of complementarity is more fundamental than the uncertainty relation because it is the reason for it. In the transformations of the unitary domain where time becomes timeless, matter becomes energy, and space becomes unextended, a teleological noumenon projects our phenomenological reality. This underlying transpiration of energy provides the "laser pump" of holonomic brain theory and provides the vehicle for integrating all aspects of QBD into one dynamic computational core - The Holonom. Its rigorous mathematical description will allow for the design of a telecerebroscope (Amoroso, 1995a), without which, no comprehensive theory of mind can hope to be complete. This allows the extracellular containment of natural intelligence presented here which could revitalize strong AI.

4 A Conscious Computer Architecture

In discussing John Searle (Searle, 1992) Henry Stapp (Stapp, 1995) states that all ontological interpretations of quantum theory "agree on the need for a dualistic ontology, with one aspect being the quantum ontology of matter, and the other aspect specifying what our experiences will be". The extracellular containment of natural intelligence in terms of a conscious computer occurs in a two level complementarity also. One is a macroscopic I/O device such as a laser system that can also solve the interface problem that has for a time held back molecular computer design. This I/O device must interface with a solid state device that has quantum effects occurring within it that mimic those occurring in the brain holoscape by producing Bose condensation.

The second component of the conscious computer is the dynamic Holonom at the core of the solid state device. The Holonom is produced by interacting tunable lasers modulated with frequencies resonant with the vacuum ground state of memory storage and retrieval at the Heisenberg matrix. The theoretical premise that memory storage and retrieval involves nonlocal processing through vacuum zero point fluctuations (Riccardi and Umezawa, 1967) suggests that consciousness pervades matter and that intelligence is a cosmological principle (Amoroso, 1996a,b). This is the general basis for a putative artificial device to embody natural intelligence. This resonance must have the ability to access the deeper nonlocal aspects of the noumenon of universal intelligence which are inherent in the fabric of spa-

cetime as a principle of nature.

The technology exists today to accomplish this feat. As further work is done in honing our understanding of the ontology, empirical work mapping out the resonances will be completed allowing assembly of the pieces (Amoroso, 1996c). "Any structure, biological or otherwise, that contained a Bose-Einstein condensate might possess the capacity for consciousness" (Zohar, 1990).

Four devices are possible with a Bose condensate core:

- 1. Telecerebroscope - Instrument for remote imaging or video recording of conscious content such as dreams and mentation. A new art form and tool for personal growth.

- 2. Conscious computer - Bose condensate core described here as the extracellular containment of natural intelligence. Production of viable personal service robots.

- 3. Psychic pacemaker - Combination of features of prior two devices but would summate predetermined spectra of "normal" or desired noetic fields as an enhancement for patient care in the psychiatric process, introducing science to the art of empathy. This device could also aid intelligence and learning.

- 4. A diagnostic device - like the reverse of the Psychic Pacemaker but instead of directing mental states would read geometries of bodymind fields fields looking for and analyzing improper stasis of Schrodinger collapse dynamics that led to diseases of consciousness or psychogenic ailments such as Alzheimer's disease or colitis (Amoroso, 1992). This bridges the gap between Eastern and Western medicine and would be the beginning of "Star Trek Medicine"

Many consciousness researchers (Hameroff, 1994, Stapp, 1995 for example) although differing slightly say quantum state reduction events are required for consciousness and use the Schrodinger wave equation to describe the evolution of the state vector potentia, the probability for each potential state reduction to occur, to describe these events. The evolution in time results in the event. Choice is the collapse of the state vector!

Noetic Field Theory (Amoroso, 1996b) goes deeper than this and suggests that the Schrodinger wave equation only describes half of the universe or half of the complementarity of consciousness. The higher level "events" described by the Schrodinger wave equation relate to events occurring in time only. State collapse is always occuring independent of thought, and doesn't necessarily require subjective consciousness or an observer. Stapp, 1995 discusses this type of collapse as occuring at the top level of consciousness. I refer to this again as only one complement occuring in the quantum brain dynamics. The other complement necessary for a conscious computer occurs at a deeper nonlocal level not described by the current quantum formalism. Quantum theory does not describe how the choice is made; this is why a deeper ontological theory is necessary to comprehend mind. Here lies the necessity for proceeding beyond the classical limit of the measurement problem inherent in the Schrodinger equation to a formulation that is outside of time. As in the EPR experiments something in nonlocality already contains the information before choice is made (Amoroso, 1996b).

This deeper more challenging aspect of the conscious computers nonlocal qualities outside of time can be accessed through the quantum potential described by David Bohm's ontology of quantum theory (Bohm, 1971). But Bohm had not gone far enough to break away from the hidden variable dogma into the new domain. Generally we apprehend only one thing at a time, one image of the possibilities of a Necker cube for example. Nonlocally all states are available simultaneously. This is the state to be produced in the core of the conscious architecture. "If enough particles occupy the same condensate, they can form a kind of giant quantum system with peculiar properties that are observable on the macroscopic scale" (Herbert, 1994).

5 Conclusion

The centuries long omission of consciousness from scientific investigation has occurred for a number of philosophical reasons; primarily the erroneous categorization of 'Res Cogitans' to the immaterial realm. Bringing it into the realm of physicality provides the potential for conscious computation. "In the history of physics where a theory dealing with one realm of phenomena, for exam-

ple thermodynamics or optics, has been reduced to a 'more basic' theory, for example statistical mechanics or electrodynamics. So why cannot psychology be likewise reduced to brain physiology, and ultimately to the basic physics of matter?" (Stapp, 1995b). How often in the history of human intellectual endeavors have the dark clouds of despair suddenly passed away with the birth of a new idea. Hopefully developing the model for a conscious architecture represents such an instance for strong AI.

References

[1] Amoroso, R.L. (1992) The Psychogenic initiation of Alzheimer's Disease *Proc. West. Psych. Assoc.* San Jose State Univ.

[2] Amoroso, R.L. (1996a) Consciousness: A Radical Definition, The Hard Problem Made Easy. *J. of Conscious Studies* In Press.

[3] Amoroso, R.L. (1996b) Noetic Field Theory: The Quantization of Mind. Forthcoming.

[4] Amoroso, R.L. (1996c) The Production Frohlich and Bose-Einstein Coherent States in Vitro Paracrystaline Oligomers Using Phase Conrtrol Laser Interferometry *Bioelectrochemistry* In press.

[5] Amoroso, R.L. (1995a) The Telecerebroscope: A Rudimentary Model. Albany: The Noetic Press.

[6] Amoroso, R.L. (1995b) The Basis of Physical Interactionism: A Material Solution for the Cartesian Mind-body Matrix. Submitted.

[7] Amoroso, R.L. and Martin, B.E. (1995) Modeling the Heisenberg matrix: quantum coherence and thought at the holoscape manifold and deeper complementarity. In K. Pribram (Ed.) *Scale in Conscious Experience: Is the Brain too Important to be Left to Specialists to Study?* Hillsdale: Lawrence Earlbaum.

[8] Bahm, G. and Pethick, C. (1991) Landau Fermi-Liquid Theory. New York: Wiley.

[9] Bohm, D. (1971) Causality and Chance in Modern Physics. Univ. of Penn. Press, Philadelphia.

[10] Dicke, R. H. (1954) Coherence in spontaneous radiation processes. *Phys. Rev.* 93: 99-110.

[11] Frohlich, H. (1983) Evidence for coherent excitation in biological systems. *Int. J. Quantum Chem* 23: 1589-1595.

[12] Frohlich, H. (1968) Long-range coherence and energy storage in biological systems. *Int. J. Quantum. Chem.* 2: 641-649.

[13] Green, M.B., Schwarz, J.H., and Witten, E. (1988) Superstring Theory. Cambridge Univ. Press.

[14] Hameroff, S. R. (1994) Quantum coherence in microtubules: A neural basis for emergent consciousness? *J. of Consciousness Studies* 1: 91-118.

[15] Herbert, N. (1994) Elemental Mind. New York: Plume.

[16] Jibu, M. and Yasue, K. (1993) The basics of quantum brain dynamics. In: K. Pribram (Ed.) *Rethinking Neural Networks* Hillsdale: Lawrence Earlbaum.

[17] Marshall, I.N. (1989) Consciousness and Bose-Einstein Condensates. *New Ideas in Psych* 7: 75- 83.

[18] Pribram, K.H. (1991) Brain and Perception. Hillsdale: Lawrence Earlbaum.

[19] Riccardi, L.M. and Umezawa, H. (1967) Brain and physics of many-body problems. *Kybernetic* 4: 44-48.

[20] Searle, J. (1992) Rediscovery of the Mind. Cambridge: MIT press.

[21] Stapp, H. P. (1995a) Why classical mechanics cannot naturally accommodate consciousness but quantum mechanics can. In K. Pribram (Ed.) *Scale in Conscious Experience: Is the Brain too Important to be Left to Specialists to Study?* Hillsdale: Lawrence Earlbaum.

[22] Stapp, H.P. (1995b) The Hard Problem: A Quantum Approach. *J. of Consciousness Studies* 2:1.

[23] Zohar, D. (1990) The Quantum Self. New York: Morrow.

Quantum Intelligence, QI; Quantum Mind, QM

Branko Souček
IRIS International Center
Via M.Troisi 18/I, 70125 BARI, Italy, fax: 0039805490290

Keywords: intelligence, quantum intelligence, quantum mind, message quantum, brain, brain-windows, generalisation, courting, mimicry, aggression, mind, behaviour, decision support systems, business systems, multi agent intelligent systems.

Edited by: Matjaž Gams
Received: May 2, 1995 **Revised:** October 10, 1995 **Accepted:** December 11, 1995

The computer-based data mining has been used to search for quantal processes. Quantizing has been observed in experimental data that come from: the frog Rana temporaria; the firefly Photuris versicolor; the brainstem auditory potentials from the human scalp. Within the frame of these experimental data, concepts of the Quantum Intelligence, QI, and of the Quantum Mind, QM, have been defined. Elementary components of QI and QM have been identified: the Optimal Quantizing; the Quantal Generalisation; the Quantum Brain Windows; the Message Quantum; the Context. QI, QM model is in excellent agreement with experimentally observed reasoning and behaviour modes: selective courting; mimicry; context switching; aggression; alternation; solo; transmitting. The relevance of these modes to the intelligent decision support business systems is shown. These fundamental modes of reasoning, behaviour, emotion, present the link between mind and computers. QI, QM leads to the new solutions for: neurological diagnoses; complex spatiotemporal data analysis and explanation; multiagent intelligent systems; brain and mind modelling.

1 Introduction

The quantal processes in neural systems have been first discovered at the neuro-muscular junction [1,3,4,6,8,9]. The quantal processes in communication and behaviour have been observed on fireflies [11,12], Katydids [10], and on the evoked potentials from human brain [19].

Recently the new, Sixth Generation Computing Techniques [7, 13 to 18] have been developed. They enhance the biological and medical data acquisition, the modelling, and the clinical diagnoses [2,5].

This work unifies previous findings and develops the Quantum Mind model, that explains the quantal brain and mind processes. In developing the Quantum Mind model, I considered the facts A to D.

A. There is a strong experimental evidence that proves the existence of the quantal processes, from the level of the neuro-muscular junction, all the way to the level of the evoked potentials and of the behaviour.

B. Natural mind is capable of solving problems of seemingly arbitrary complexity. Yet its learning procedures seems to follow simple principles, such as the Hebbian rule.

C. Psychological experiments suggest the information flow of 1011 bits per second for human sensory input, but only about 10 bits per second for the input into short term memory. It is clear that enormous data compression, clustering, quantizing takes place.

D. The conversion of an continuos signal into discrete descriptors necessarily involves quantizing. The quantizing performs grouping or round off the continuos signal into groups or classes. The fundamental concept of quantizing is the subject of the quantizing theorem [14].

Material and data: The KNOWLEDGE MINING process in a computer forms and modifies

the hypotheses until a pattern emerges. In this way it extracts the relations hidden in the experimental data. The data come from three data bases. END PLATE: the end plate potentials were recorded from the median extended longus-digitorium IV muscle of the frog Rana temporaria. INSECT: the fireflies Photuris versicolor were courted using artificial flashes of duration between 0.1 and 0.2 seconds. HUMAN: the brainstem auditory evoked potentials were obtained from the Vertex-left mastoid, Vertex-right mastoid electrode locations on the scalp.

In this work I recognise the group of processes, that I call the quantal processes : the transmitter release; the Brain-Windows communication; the evoked potentials from human brain.

Out of these processes I extract the fundamental, elementary components: the Optimal Quantizing; the Quantal Generalisation; the Quantum Brain Windows; the Message Quantum; the Context. This list is open for further investigation.

I present here only contribution of the Quantum Brain Windows to the Quantum Mind concept. For details, and for other quantal process, see [13,19].

2 The Quantum Brain Windows

I take the Brain-Windows concept from our biological experiments [11,12,13,19].

The Brain-Window is defined by its receive R side and its send S side. A transition from continuos, fuzzy signalling to discrete coding is achieved and a communication language is formed. Both receive and send windows are adaptive and also depend on the context.

In the nature the response [latency L] is a continuos analogue function of the stimulus [interval I], Figure 1. The basic, quantal law of brain windows is:

- If there is a match between the stimulus [interval I] and one of the receive windows, and
- If there is a match between the internally induced response [latency L] and one of the sending windows,
- Then the response [latency L] will be send out.

Note that the brain windows define a quantal, fuzzy, symbolic presentation. The I/L transfer functions (belts b, d in Figure 1) define the stimulus response relation. The brain-window mechanism uses continuos and discrete information, connected through the quantizing process. The experiments [11,12,19] show the existence of brain-windows in the external sensory and motor layers. Further experiments would be necessary to search for a possible existence of brain windows quantizing process deeper in the neural system.

The experimental data and computer model-generated data show that the latency interacts with the sensory inputs and with behaviour in several ways:

1. The latency L is a continuos [analogue] function of the stimulus interval I, and of the content of the memory, as shown with belts b, d, in Figure 1.
2. The latency L is also a discrete [discontinuous] function of the context stored in the memory. In this way, the latency and answer can be switched from one window to another, although the windows are far away.
3. Interaction between belts b, d, and the windows is equivalent to the process of quantizing. The process of quantizing the information is necessary to form the basic messages of the language.
4. The windows are discrete but adaptive. The windows change shape, depending on the behaviour and on external stimulation. By narrowing the window, the animal becomes highly selective during courting communication. By widening the window, the animal increases its chance of catching the prey during aggressive mimicry.
5. The brain-windows mechanism uses continuous and discrete information, connected through the quantizing process. The experiments presented here show the existence of one layer of continuos, quantizing, discrete [CQD] information. This layer is the easiest to investigate, because it involves the external sensory and motor logic. It is expected that additional CQD layers exist deeper in the neural system.

The brain windows quantizing concept could be used to establish a common language to communicate between different members of a multi- agent

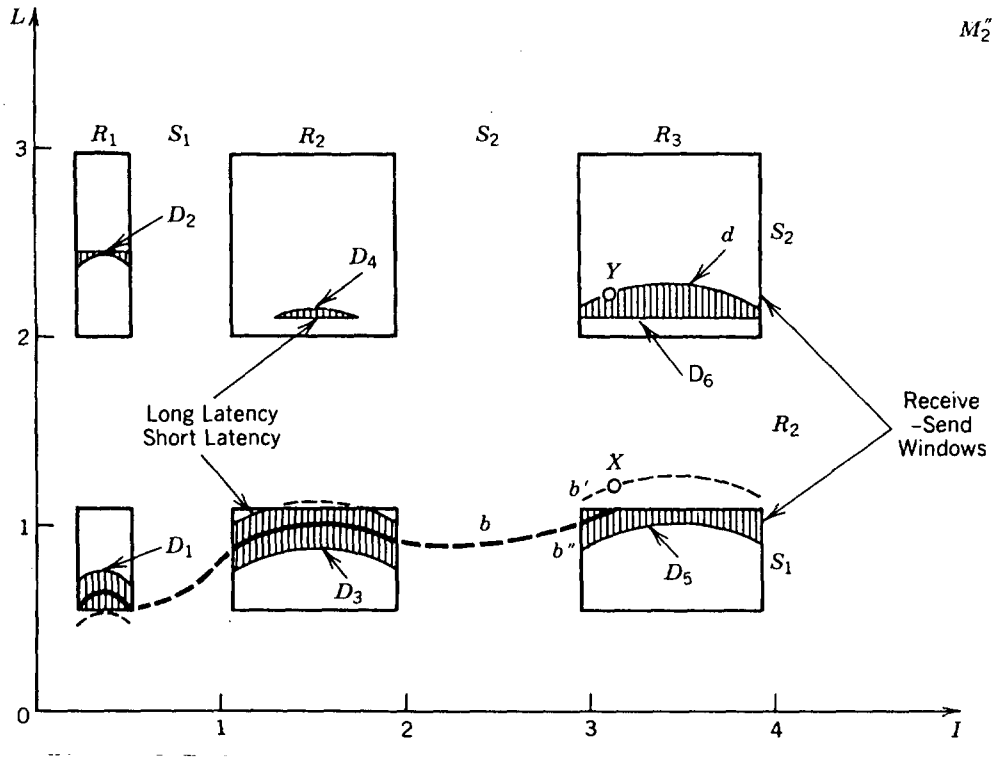


Figure 1: Brain-Windows language of the firefly. The language is formed of dialects [schaded area]; dialects are intersection between continuos belts [b, d] and discrete windows.

system. In other words, it is could be used to create a new kind of intelligent systems.

The Brain-Windows serve for communication between QIs. Upon receiving the stimulus, QI generates a set of receive and send windows. Each receive window (vector) is related to specific behaviour. QI will accept the stimulus only if it matches one of the receive windows; in this way the QI understands the stimulus context. QI will answer through one of the send windows (vectors) and in this way sends back a meaningful information.

3 The Quantum Mind

The Quantum Mind concept readily explains several modes of reasoning and behaviour [10,11,12]. I compare these behaviour modes with the behaviour in the human society, and in particular in business. Detailed description of the biological examples is given in [13]. Related adaptive, intelligent business systems are presented in detail in [7]. Based on incremental learning and on interaction with the user and the environment, the QM system modifies its parameters and in this way changes its behaviour, in the following way.

Selective courting. The QM system gradually narrows the windows. The goal is to detect/select the right partner and to avoid the risk of courting the wrong partner.

Biology: brain windows before mating.

Business: low risk credit scoring.

Mimicry. The QM system gradually widens the windows. The goal is to attract/select as many partners as possible.

Biology: brain windows for feeding.

Business: help desk for marketing.

Context Switching. Sudden change in QM behaviour, based on the past history, recalled experience and on environmental conditions.

Biology: brain windows after mating.

Business: EDI-switch for adaptive purchasing.

Aggression. One subsystem tends to take control of the whole QM system.

Biology: time coding in insect chirping.

Business: competition networks.

Alternation. Two [or several] subsystems are taking control of the QM system in alternation.

Biology: time coding in insects; leader/follower chirping.

Business: travelling salesman, genetic programming.

Solo. One subsystem is in control of the whole QM system.

Biology: time coding in insects; leader chirping.

Business: winner takes all.

Transmitting. Exchange of quantized messages among subsystems.

Biology: quantal transmitter release on neural terminals.

Business: standard message packages.

The quantal processes involve applying the experience as represented in past similar situations, to analyse and solve current problems. I consider the quantal processes to be an important factor in reasoning and in the control of the behaviour and of the society.

4 From the Quantum Mind of the Femme Fatale to the Intelligent Multi-Agent Business System

The hierarchical model presents the basis to build a sensory-processing goal-directed system. The multilevel, multivariable hierarchical feedback system explains many features of living systems, including: goal-seeking as the natural form of behaviour, sensory processing hierarchy, pattern recognition, internal world representation and the functions found only in higher animals and man.

Sensory data enter this hierarchy at the bottom and are filtered through a series of sensory- processing and pattern-recognition modules arranged in a hierarchical structure that runs parallel to the behaviour-generating hierarchy. Each level of this sensory-processing hierarchy processes the incoming sensory data stream, extracting features, recognising patterns, and applying various types of filters to the sensory data. Information relevant to the control decision being made at each level is extracted and sent to the appropriate behaviour generating modules. The partially processed sensory data that remains is then passed to the next higher level for further processing.

In figure 2 I present the multi-level Quantum Mind model of the Femme Fatale firefly. I explain our experimental findings [11,12] in the following way: Each level of behaviour generating hierarchy receives the command C from a higher level. It also receives the feedback F from the environment.

The output from the operator H selects one of the possible subcommands on the next lower level.

For example, the level "SURVIVAL" selects the subcommand for the next lower level from the set $C_2(C_2', C_2", C_2"')$. Which subcommand is selected depends on the feedback vector F_3. In other words, $C_2 = H_2(F_3)$. Similarly the level "REPRODUCTION" selects the subcommand for the next lower level: $C_1 = H_2(F_2)$.

When the hormone level and blood chemistry indicate the proper time, and the air temperature is right, the command C_2 is selected. When $C_2 = C_2'$ indicates reproduction, and F_2 indicates that external stimuli are present in the form of light flashes, and the male is in the territory, the command C_1 is selected. When $C_1 = C_1'$ indicates "PATROLLING FLASHING", motor control, internal oscillator, and the lantern, execute this command.

Both commands and feedback are coded in a slow scale [chemical hormonal coding] and in a fast scale [pulse/time coding]. Pulse/time coding is of special interest for behavioural patterns that are executed as time sequences of events. The quantal Brain Windows are used in such sequences, to screen, check, or recognise the information.

Similar QM concept could be used to build intelligent multi-agent business systems.

5 Conclusion

The presence of the quantal processes in experimental data is not directly visible. To recognise the quantal process, I have developed a semiautomated knowledge mining procedure. Using this procedure I was able to recognise and to analyse three quantal processes. The quantizing is present in the following way.

a) The transmitter release, is built up from m.e.p.p. quanta, in the range of 0.2 to 0.8 mV; quantizing enhances the small generalisation.

b) The Brain-Windows, are built up from message quanta of 0.1 to 1 s; observed in communication

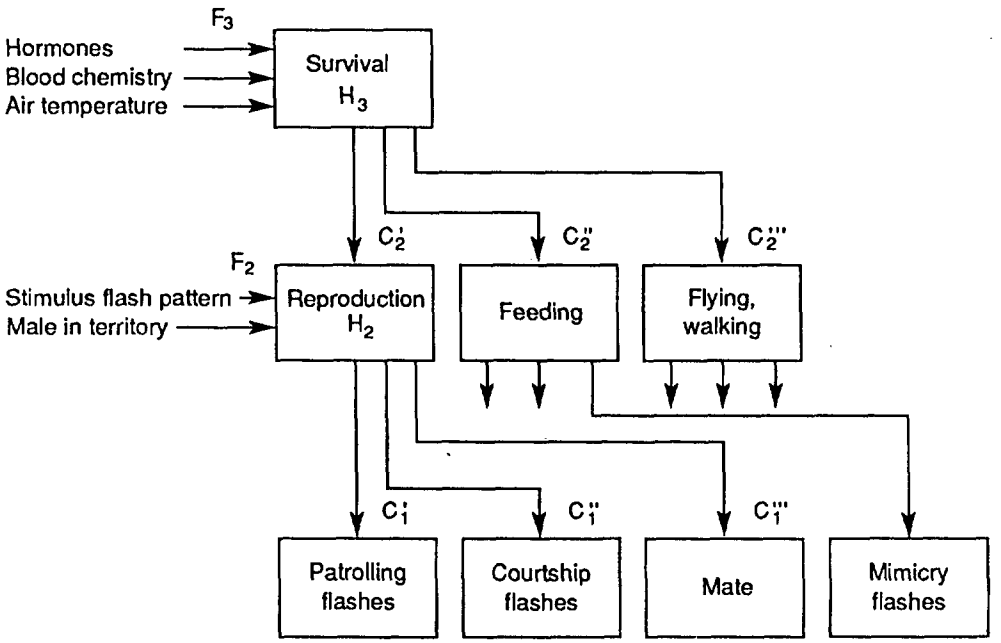


Figure 2: The Quantum Mind model of the Femme Fatale. The control hierarchy for the female firefly.

and behaviour; quantizing enhances the large generalisation, resulting from the collective action of a large population of neurones.

c) Evoked potentials; experimental data reveal the presence of quantal processes; their origin and function is not yet clear.

I have identified several elementary components, that are relevant for biological quantal processes: the Optimal Quantizing; the Quantal Generalisation; the Quantum Brain Windows; the Message Quantum; the Context. As these components operate in a concert, their functions overlap.

The components operate within their quantizing, clustering, generalisation spaces. The spaces cover local similarities, as well as global features. **Results:** Identified elementary quantal components: The OPTIMAL QUANTIZING extracts the most meaningful information. THE QUANTAL GENERALISATION merges the quantal representation of inputs with the stable converging. The QUANTUM BRAIN WINDOWS define the communication language of the intelligent system. The Receive windows are related to the legal, acceptable stimuli or question messages. The Send windows are related to the legal, acceptable responses or answer messages. Both windows are adaptive and correlated with the behaviour. The MESSAGE QUANTUM is the smallest information item used in coding. It is related to the primary biological oscillator. The CONTEXT depends on the situation and on the conditions, such as the hormone level, blood chemistry, body and air temperature.

The neural and behavioural quantal processes have been discovered more than twenty [1, 3, 4, 6, 8, 9,], ten [10, 11, 12] and one [19] year ago. **These experimentally proven quantal processes support important functions in reasoning, behaviour, emotion, mind.**

6 Discussion

The quantal transmitter release has been first observed at myneural junctions of frog muscle. It has been observed also on mammalian muscle and has been proven on a number of other preparations.

The quantal communication and brain windows have been first observed on insects: katydids and fireflies. These are quantal processes in time axis.

The time axis is a perfect information carrier. Time intervals are noise and drift immune, and they cannot be attenuated or amplified. The behaviour of living systems is naturally programmed into the time axis as a sequence of events.

The sequence is controlled by neural oscillators or pacemakers. The pacemakers are sensitive to sensory inputs which could turn them on or off.

In the case of fireflies, the first half-period of the oscillator is the shortest one: it is approximately 0.25 sec. This is one-eight of the period of the nonmodulated primary waveform ($T_1/8$). As this is the smallest information item used in coding, I call it the Message Quantum, q: $q = T1/K$ (in the case of the firefly, K=8)

I use the Message Quantum q to measure the windows: For $q < R1 < 3q$, the window width is $2q$. For $5q < R2 < 9q$, the window width is $4q$, and so forth.

The evoked potentials from human brain are used for both, the brain research and the clinical diagnoses.

Brainstem Auditory Evoked Potentials (BSAEPs) are generated in response to a brief auditory stimulus with seven peaks appearing within 10 ms following the stimulus in normal subjects. Pathological states resulting from head injury, acoustic tumours and multiple scleroses give rise to delays in the transmission of electrical signals and consequently the peaks are abnormally located.

It is believed that the BSAEP latencies provide necessary and sufficient information to discriminate between normal and pathological states. The experiments have shown that the 2nd, 3rd and the 4th peak latencies are the optimal features for classification [2,5].

The seven distinct peaks suggest that BSAEP is a quantal process. Furthermore, I have noticed a similarity, between the recorded BSAEPs waveforms and the Brain-Window waveform. Hence I follow the Brain-Window concept, to explain the human evoked potentials.

More than twenty years of experience with the neural and behavioural Quantal Intelligence tells me, that quantizing, clustering, information compression is present in several domains: amplitude, time, frequency, field, electrical, chemical, hormonal.

References

[1] Boyd I.A., and Martin A.R., 1950, J. Physiol, London, 132, 74.

[2] Chiappa K.H., 1989, Evoked Potentials in Clinical Medicine, Raven Press Ltd, New York.

[3] Del Castillo J. and Katz B., 1950, J. Physiol, London, 124, 560.

[4] Del Castillo J. and Katz B., 1952, J. Physiol, London, 125, 546.

[5] Ho R., Sutherland J.G., Bruha I., Neurological Fuzzy Diagnoses: Holographic vs Statistical vs Neural Method, in Ref. [7].

[6] Katz B. and R. Miledi, 1965, Proc. Royal Soc., London, Ser. B, 161, 483.

[7] Plantamura V.L., Soucek B. and Visaggio G., 1994, Frontier Decision Support Concepts, Wiley, New York, pp.401.

[8] Soucek B., 1971, Complete model for the statistical composition of the end-plate potential, Journal of Theoretical Biology vol 30, pp.631–645.

[9] Soucek B., 1971, Influence of the latency fluctuations and the quantal process of the transmitter release on the end-plate potentials amplitude distribution, Biophysical Journal vol 11, No. 2, pp.127–139.

[10] Soucek B., 1975, Model of alternating and aggressive communication with the example of Katy-did chirping, Journal ot Theoretical Biology vol 52, pp. 399–417.

[11] Soucek B. and Carlson A.D., 1986, Brain Windows in Firefly Communication, Journal Theoretical Biology vol 119, pp.47–65.

[12] Soucek B. and Carlson A.D., 1987, Brain Windows Language in Fireflies, Journal Theoretical Biology vol 125, pp.93–103.

[13] Soucek B. and Soucek M., 1988, Neural and Massively Parallel Computers, Wiley, New York, p.460.

[14] Soucek B., 1989, Neural and Concurrent Real-Time Systems, The Sixth Generation, Wiley, New York, pp.387.

[15] Soucek B. and IRIS Group, 1991, Neural and Intelligent Systems Integration, Wiley, New York, pp.664.

[16] Soucek B. and IRIS Group, 1992, Fuzzy, Holographic, and Parallel Intelligence, Wiley, New York, pp.330.

[17] Soucek B. and IRIS Group, 1992, Dynamic, Genetic and Chaotic Programming, Wiley, New York, pp.650.

[18] Soucek B. and IRIS Group,1992: Fast Learning and Invariant Object Recognition, Wiley, New York, pp.279.

[19] Soucek B., 1994, Neurological Diagnoses Based on Evoked Brain-Windows and on Holographic Learning, Informatica vol 18, pp.109–114.

Representations, Explanations, and PDP: Is Representation-Talk Really Necessary?

Robert S. Stufflebeam
Washington University, Philosophy-Neuroscience-Psychology Program
Campus Box 1073, One Brookings Dr., St. Louis, MO, 63108, USA
Phone: (314) 935-6670, Fax: (314) 935-7349
E-mail: rob@twinearth.wustl.edu

Keywords: representation, computation, discovery, explanation, PDP

Edited by: Marcin Paprzycki

Received: May 12, 1994 **Revised:** November 26, 1995 **Accepted:** December 5, 1995

According to the received view, since the brain is a computational device, "internal representations" need to figure in any plausible explanation for biological computational processing. My aim here is to show that such is not the case: 'internal distributed representations' can be dropped altogether from mechanistic explanations of parallel distributed processing [PDP]. By focusing on the discovery of mechanistic explanations for complex systems [sections 2-3], I argue that PDP networks cannot be functionally decomposed into component internal distributed representations. I also argue that 'distributed representations' are not internal representations, but rather constructs [section 4]–interpretations imposed on the processing. So, if the brain is a PDP-style computer, then there are reasons for thinking that internal representations are not doing the work they are commonly thought to do.

1 Introduction

Many disciplines are engaged in the naturalistic attempt to explain and model cognition. Notwithstanding the interdisciplinary character of the undertaking, nearly every theory of cognitive processing is built upon the assumption that the brain is a computational device – a computer. And since "computation presupposes a medium of computation" (Fodor, 1975, p. 27), it follows that the brain is a representational device – "a device that has states or contains within it entities that are representations" (Von Eckardt, 1993, p. 143). Thus, about this much at least, almost everyone agrees: "at the heart of a scientific understanding of cognition lies one crucial construct, the notion of internal representation" (Clark & Toribio, 1994,401).[1]

Still, what one takes to be an internal representation depends on whether one is a *classicist* or a *connectionist*. For classicists (e.g., Newell & Simon,1972; Fodor, 1987; Fodor & Pylyshyn, 1988; Cummins, 1992), internal representations are the explicitly symbolic, quasi-linguistic structures that mediate the combinatorial, rule-following production of a system's output. For connectionists (e.g., Rumelhart et al., 1986; Churchland & Sejnowski, 1989; Elman, 1992), internal representations – called 'distributed representations' in networks implementing parallel distributed processing [PDP] – are distributed patterns of activation among the processing units, patterns that while not explicit symbolic structures, nevertheless figure causally in the production of a network's output.

Herein lies the problem: The *raison d'être* of neuroscience is to determine how brains work, so naturalistic explanations of cognitive proces-

[1]There are other reasons for thinking that internal representations need to figure in explanations of cognititive processing. For example, it is assumed that while the behavior of reactive, stimulus-driven creatures need not require conceptualization and internal representation, intentional behavior surely does: action management, desire management, planning, and a host of other cognitive activities cannot occur in the absence of internal representations (Kirsh, 1991).

complex localization is possible, then the system exhibits what they call *first order interaction.* To complete the mechanistic explanation, like before, one must continue the functional analysis to the subcomponents. This time, however, particular attention must be paid to the interactions among components.

If neither direct nor complex localization is possible, then, **STEP 6**, determine whether the phenomenon or problem under study needs to be *reconceptualized.* For example, findings from other disciplines may place unanticipated constraints on the problem, constraints that force one to reconstitute the phenomenon and begin again.

If the phenomena under study doesn't warrant reconceptualization, then because decomposable systems are modularly organized, if a system isn't modular, it can't be decomposed (p. 24). Moreover, since localization entails a "realistic commitment" to the f unctions isolated in the task decomposition, if appropriate techniques fail to show *which* components are performing these functions, then the functions can't be localized (Bechtel & Richardson, 1993, p. 25). If decomposition and localization are *not* possible [and assuming that the problem has been conceptualized correctly], then the system is fully integrated and non-decomposable. END. [See **Fig. 1**.]

3 Types of complex systems

Bechtel & Richardson (1993) cite many examples of the heuristically driven mechanistic discovery program in action. In so doing, they reveal a host of ways complex systems can be organized, from the fully decomposable to the non-decomposable. But where along that continuum should connectionists systems be placed? [See **Fig. 2**.] If the reason connectionist systems do what they do lies *not* in the components but rather in the way the components are organized, then decomposition should not be possible. Historically, however, forgoing decomposability required abandoning the search for a mechanistic explanation (Bechtel & Richardson, 1993, p. 199). Thus, is it possible for connectionists to forgo decomposition and yet provide a mechanistic explanation of PDP? If so, then connectionist systems represent an alternative way of elaborating a mechanistic program (Bechtel & Richardson, 1993, p. 199). But if connectionist systems are *not* decomposable, then how can one localize specific systemic functions to discrete distributed representations?

3.1 Are PDP networks decomposable?

In the structural sense, *of course* PDP systems are decomposable: they decompose into layers, processing units, and connections. The question, however, whether we can functionally decompose such systems, and if so, can we then localize any of the network's functions to any of the network's 'distributed representations', components, or collections of components? My plan here is to first sketch-out how PDP works. I shall then apply the above decision procedure to a PDP network. In the end, I argue, PDP systems are not decomposable in the requisite sense, though it *is* possible to give a mechanistic explanation for how they do what they do. First things first.

3.1.1 How does PDP work?

There are several types of connectionist architectures, not all of which implement PDP. And the variety of PDP architectures preclude my going beyond identifying a few general features of PDP in a rather simple, though idealized, feed-forward, three-layered network. Although the processing in such nets can [sometimes] be described in terms other than PDP, say, 'localist representation', only PDP produces paradigmatic distributed representations. Or so it is claimed.

PDP – or 'distributed representation' in the processing sense – occurs in a task-specific network of interconnected units, whose arrangement, following training, make the network capable of performing a complex task, yet without the need for mediating symbols or even explicit rules. Above all else, *nonsymbolic* [analog] *computation* and *distributed encodings* are what distinguish PDP from every type of classicist representational processing. For example [see **Figure 3a**], after receiving an input pattern, each input unit computes an activation value as output, say, some number within a continuous range from -1 to 1. This activation value is then transmitted via connections – called *weights,* which vary in strength from unit to unit – to each of the units in the succeeding *hidden unit layer.* Each of the hidden units then computes its activation value,

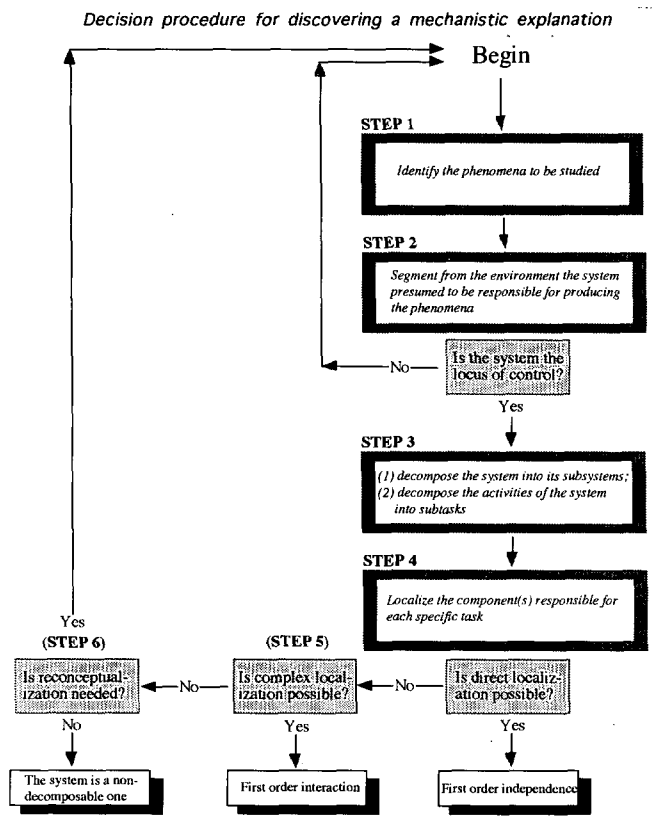


Fig. 1: Two further points are worth noting. First, collateral theories and other constraints play a key role in narrowing the problem space. Second, constraints discovered at a lower-level may force one back to a previous step, or even to the beginning.

which, typically, is determined by summing over both (a) the *products of the activation values* of each unit to which the target is connected and (b) the *weights* associated with each such unit [see **Figure 3b**]. The activation value of each hidden unit is then similarly propagated via its connections to each of the units in the output layer, which then compute their activation values.

Now comes the interesting stuff. Suppose the system depicted in **Fig. 3a** is a rather ordinary – but successful – pattern recognition network [a task for which PDP networks are particularly well suited]. The 'objects' that the network recognizes fall are tokens of the following three motorcycle types: Harley, Yamaha, and Honda. Of course, the network isn't *really* able to recognize tokens of any of these motorcycles. Rather, what it *can* do, let us suppose, is associate a given input array with a 'name' – the target output-that corresponds to one of these three types of motorcycles: Harley 1, 1, 1, Yamaha -1, 0, -1, and Honda 1, -1, 0. Here is how it learns: During training, the network is presented again and again [for n number of epochs] with a distorted set of the prototypical input arrays that correspond to each type of motorcycle. Then, after each epoch, the random initial setting of the system's weights is gradually adjusted via a learning algorithm: some connections are increased, others are decreased. To test the network following training, the system is presented [for the first time] with the prototypical input arrays that correspond to each of the three motorcycles. If the test is successful, then the network's output pattern will correspond [within some accepted range] to the target outputs.

Note that in PDP, the encodings necessary to complete the task are extended over many processing units. That is, each item of interest "is represented by a pattern of activity distributed over many computing elements, and each computing element is involved in representing many different entities" (Hinton et al., 1986, p. 77). As such, each unit can have a value of 1 for *more* than one item of interest, and different items of interest can be stored as patterns of activity over the *same* set of units (van Gelder, 1991, p. 43). Thus, a distributed, subsymbolic manner of processing and storing information is – to paraphrase Clark

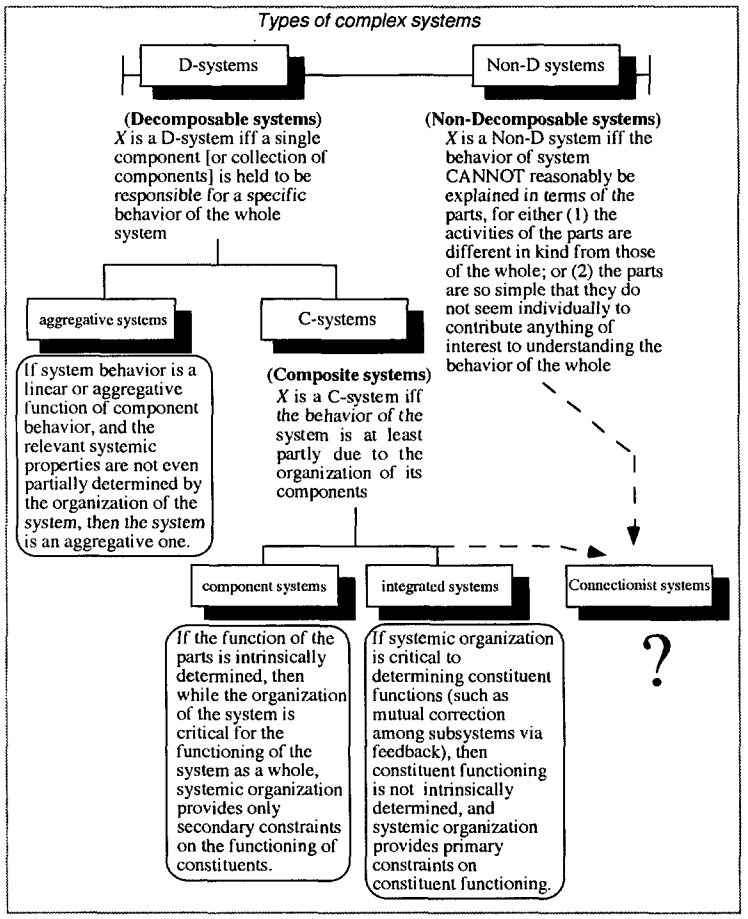


Fig. 2

(1989) – a very natural, fast, and relatively cheap way of achieving what classicist or localist processing does, but with far fewer processing units.[2]

3.1.2 Is a PDP network functionally decomposable?

While a connectionist approach represents a radical alternative to rule-governed, explicitly symbolic computational processing, it isn't obvious whether the approach provides an alternative to the decomposition-dependent mechanistic program. To resolve this issue, and with the above pattern-recognition network in mind, let us try to complete the decision procedure depicted in **Fig. 1**.

STEP 1 *Identify the phenomena to be studied.* ANSWER: Pattern recognition.

STEP 2 *Segment from the environment the system presumed to be responsible for producing the phenomena.* ANSWER: OK, there's the network. QUESTION: *Is the system the locus of control?* ANSWER: Yes.

STEP 3 [Part 1] *Decompose the system into its subsystems.* ANSWER: The network has three layers, each composed of a small number of processing units: the input and output layers each have three units; the hidden layer has five units. Each unit in the lower layer is connected to each unit in the succeeding layer. [Part 2]: *Decompose the activities of the system into subtasks.* ANSWER: There are three tasks: the Harley association task, the Yamaha association task, and the Honda association task.

So far so good. Now the problems begin.

STEP 4 *Localize the component(s) responsible for each specific task.* QUESTION: *Is direct localization possible?* ANSWER: No. The activity in no one layer is responsible for any of the tasks that we want to explain. Nor is it the case that any one processing unit is responsible.

STEP 5 QUESTION: *Is complex localization possible?* ANSWER: No, for the same reasons as above.

[2]For more on the architecture of PDP, see Bechtel & Abrahamsen (1991), Clark (1989), or Rumelhart et al., (1986); cf. Fodor & Pylyshyn (1988) and Pinker & Prince (1988).

Feed-forward, three-layered network

Fig. 3a

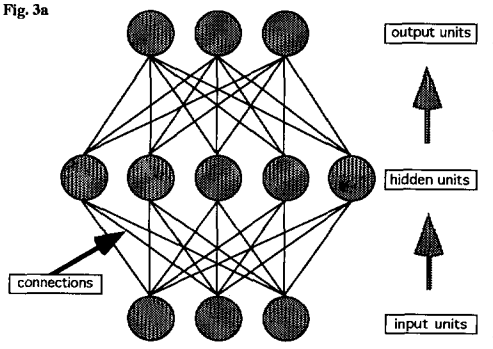
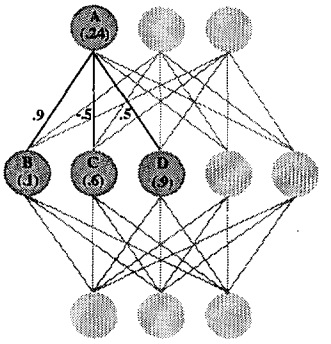


Fig. 3b



Again, the activation values for this network vary over a continuous range from -1 to 1. To determine the activation value of A, let us suppose that A is the target only of units B, C, and D from the hidden unit layer. "The input to unit A is determined by multiplying the activation of B, C, and D by the weights connecting them to unit A, and then adding those values; this yields a net input to unit A of .24. The activation of A may then be determined by a variety of formulas. Some take the prior activation of A into account; some do not" (Bechtel & Richardson, 1993, p. 212).

STEP 6 QUESTION: *Is reconceptualization [of the phenomenon under study] warranted?* ANSWER: No, for the behavior we want to explain does indeed exist. Moreover, there is no question that the above network is indeed the system responsible for producing the phenomena.

Because the components of such PDP networks perform no recognizable subtasks (Bechtel & Richardson, 1993, p. 222), nothing less than the dynamic properties of *the entire system* are sufficient to explain the phenomena under study. As such, PDP systems are not decomposable in the requisite sense. Does it follow, therefore, that one cannot provide a mechanistic explanation for how such systems do what they do? Of course not. Indeed, I have already done so [abridged, granted, but an explanation nevertheless]. Thus, PDP-style, nondecomposable complex system *do* represent an alternative to the traditional mechanistic program.

Given the biological plausibility of nonsymbolic [analog] processing, it is a virtue of PDP that it permits certain types of cognitive behavior to be mechanisticly explained as *an emergent property* of a complex system, rather than by having to posit mediating internal representations. But since as yet I have made no reference to distributed representations, lest I be accused of being too hasty, let's consider whether I have omitted a necessary part of the explanatory story.

3.2 *Describing* PDP vs. *Explaining* PDP

Can internal representations figure in mechanistic explanations? Of course. Indeed, such references would be required if the system in question were a classicist one: digital computation is necessarily dependent on internal symbolic representations. But connectionist networks are not classicist systems. Hence, why expect that internal representations *should* figure in explanations of PDP [or analog] computation? There are two reasons.

First, forgoing decomposability has historically entailed abandoning the mechanistic program. If distributed representations are full-blooded internal representations, then it is reasonable to treat distributed patterns of activity as spatially distributed components of the system, compo-

nents that can be identified with specific subtasks. Thus, both decomposability and the mechanistic program can be preserved.

But PDP networks aren't functionally decomposable. Indeed, "connectionist models explain performance without explicitly or necessarily decomposing that performance into intelligible subtasks" (Bechtel & Richardson, 1993, p. 221). Moreover, distributed patterns of activation are not components of the system; rather, they are an emergent byproduct of the network's *representation-in, representation-out* processing.[3] In the network described above, for example, each prototypical input pattern and each target output pattern is a 'name', of sorts, for each of the three types of motorcycles. Are such patterns arbitrary? Yes. Do they stand for something? Yes. Does biological processing require such constructs? Of course not. The point, however, is that while systemic inputs and outputs are representations, the processing itself, however, isn't. Since I have already shown that the mechanistic program need not be abandoned if decomposability isn't possible, it simply isn't necessary to appeal to internal representations when explaining PDP systems – save, of course, the inputs and the outputs. Why should connectionists find this bridling? They already treat PDP as a fundamentally different sort of representation producing processing. What they also need to recognize is that PDP is *so* different, it need not be explained in the internal representation-laden manner demended for classicist processing. Not everything that goes on in the production of representations need be representations themselves. For it to be otherwise, representation-talk would loose its explanatory efficacy.

Second, there is no gainsaying that distributed patterns of activation among the processing units can be treated *as* if they are components. As such, because one *can* construe each such 'component' as fulfilling some part of the network's overall pattern recognition task, one can also describe the system's behavior in a representation-laden way.

The problem, however, is that descriptions are not explanations. Explanations, recall, require a realistic commitment to the entities being posited. For the time being, ignore my having shown that distributed representations are not components of PDP systems [which, by the way, makes them poor candidates for localizing systemic functions]. Are distributed representations appropriate objects of a realist stance? If so, then would be at least *some* reason for appealing to internal representations in explanations of PDP. Don't hold your breath.

[3]Though the input and output patterns are representations, they aren't what connectionists are referring to by the term 'distributed representations'. I shall have more to say about this presently.

4 Are distributed representations *really* representations?

If so, then references to internal representations would at least be justified [though not necessary] when explaining a given PDP network's behavior. If not, then independent of whether a connectionist system is decomposable, use of internal representation-talk for explaining PDP would be unwarranted.

Most connectionists are revisionists; i.e., they treat distributed patterns of activation among the processing units as a fundamentally different sort of representational entity, though representations nevertheless. But what is it about distributed patterns of activation that compels connectionists to call them *representations*?

4.1 Is a distributed representation simply any state of PDP?

Is what makes a distributed pattern of activation a distributed representation simply the fact that it is a state or a product of PDP? Clearly some connectionists think so, though few are quite so explicit about it as Hatfield (1991): "What makes a state a representation is the fact that it is a state of a system whose function is to generate representations" (also see pp. 171-173). Since the function of PDP networks is to produce representations – via the association between content-bearing inputs and target outputs – any state of a PDP network would be a representation. Thus, it would seem, to be a 'distributed representation' is simply to be any state of a network implementing PDP – parallel distributed processing.

For the following reasons, however, distributed representations are *not* simply states of a network

implementing PDP.

First, PDP does *not* produce all and only distributed representations. For example, 'NETtalk' is a PDP network that transforms text input into phonemes (Churchland & Sejnowski, 1989). Even if one wanted to say that phonemes are representations, do phonemes really count among the sort of things connectionists [or anyone else] want to call 'distributed representations'? No.

Second, if the phonemic output of NETtalk counts as a representation, then not every representation that may be ascribed to [or located in] a PDP network is a distributed representation.

Third, if simply being a state of PDP made that state a distributed representation, then representations would be "so unconstrained as to be without content" (Hatfield, 1991, p. 167). As such, representation-talk would loose its explanatory efficacy. Thus, if simply being a state of PDP made that state a distributed representation, then the meaning of 'a distributed representation' would be both trivial and uninteresting. But given how often connectionists appeal to "internal representations" (Elman, 1992, p. 138) in their 'explanations' of connectionist systems, it cannot be assumed that connectionists are using 'distributed representations' in a trivial fashion.

But if connectionist are not using the term 'distributed representations' in a trivial fashion, then there has to be *something* about distributed patterns of activation that warrants their inclusion in the larger class of things we call representations. Thus, again, what is it about distributed patterns of activation that compels connectionists to call them *representations*?

4.2 Representational pluralism

If one feels that it is for connectionists to decide what it is that makes a distributed pattern of activation a representation, then one will probably agree with the following defense of representational pluralism: The semantics of the term 'representation' varies considerably from theory to theory. Thus, it would be "naive" to expect members of competing theoretic camps to "use of the same notion of representation," and quixotic to delve into "the nature of representation" independent of any particular theory (Cummins, 1989, pp. 12-13; Stich & Warfield, 1994). Hence, if one's ontology is fixed by one's theory, and connectionism entails that there are distributed representations in a PDP network – or, if connectionists wish to stipulate that a distributed representation is simply any state of PDP – then, at least for connectionists, there *are* distributed representations. Simply put, so the argument goes, it is for connectionists to decide what they want to call a representation.

I remain unconvinced, particularly since *I'm* a connectionist. First, it doesn't follow that there *ought* to be multiple conceptions of representation simply because there *are*. Indeed, I would argue that transdisciplinary discourse about cognition is obstructed by the multiple conceptions of representation. Second, from stipulative practices or entailments, it does not follow that distributed representations actually exist [remember *phlogiston*?], much less that the concept of internal distributed representations is coherent. Last, the above defense hardly avoids the charge that connectionists are using the term 'a distributed representation' in a trivial fashion.

Thus, if distributed representations are indeed representations, they must meet the general identity conditions common to all representations, distributed or otherwise. "What are these conditions?" I'm glad you asked.

4.3 What makes a representation a representation?

Except where the term 'representation' has been utterly trivialized, it means either (1) *the* **process** *of representing* – to stand for, to symbolize, or to depict some other thing [or event]; or (2) *the* **thing** [or state] *that stands for* [symbolizes, or depicts] *some other thing* [or event]. Regardless of the obviousness of this *process-thing distinction*, writers who use the term 'representation' often not only fail to make explicit which sense they have in mind, they frequently employ both senses in the same context.

A related but far more serious confusion obtains when one treats the production of representations as itself *a representation*. For example, assume there exists a finite state grammar G for a finite state language L. Because there exists an L such that G stands for L and $(G \neq L)$, according to the above definition, G would be a representation (Chomsky, 1957). Strictly speaking, however, G stands for the mechanism [or "algorithm"] that produces L, not L itself (cf. George, 1989; Pea-

cocke, 1989). But such a *process* is exactly what G in fact is, so (G=L). As such, there is also a sense in which G isn't a representation. This conflict arises because of the ambiguities attending the above *process* sense, which can mean either (a) *the state of standing for something else*, or (b) *the mechanism by which representations are produced.* Although both "the state" and "the mechanism" are things that may legitimately be called 'representation', only the former nontrivially captures the 'thing' sense of *a representation.*[4]

To avoid the ambiguities attending the unconsidered use of such terms as 'representation', 'mental representation', 'distributed representation', and the like, unless explicitly stated otherwise, I construe 'representation' only as a *thing* [or *state*] that stands for [symbolizes, or depicts] some other thing. To disambiguate the *process* sense, I use 'is in a stands-for relation' to capture what is ordinarily meant by the term 'represents'. And I reserve the term 'process' for a mechanism by which representations are produced.

4.3.1 Individuating representations

'R' below not only reflects the above distinctions, it seems to capture in a general, pan-theoretic way, what is necessary and sufficient for something to be a representation: $\mathbf{R} = X$ is a representation if and only if

1. there exists a Y such that X stands for Y;[5]
2. X is not a representation-producing process; and
3. $(X \neq Y)$.

The following two reasons make **R** a compelling principle for the individuation of representations.

First, representations are posited in many theories, not just those pertaining to cognitive processing. For example, *the Mona Lisa*, a *mental image* of the Mona Lisa, and *the concept* 'Mona Lisa' are each, in their respective theories/disciplines, *a representation.*[6] More importantly, with **R** as the standard by which to individuate representations, they would remain so. This would *not* be the case, however, if – following Fodor (1987) – **R** restricted the things that may be representations to only "symbols and mental states." Hence, **R** avoids circumscribing representations too narrowly.

Second, because 'a representation' is *not* shorthand for *an internal representation*, much less *a mental* one (cf. Cummins, 1989; see Sellars, 1980, 15), according to **R**, states in or products of *non-cognitive* systems could thus be representations.[7] Since PDP networks are not yet themselves cognitive systems, question-begging against connectionists is thus avoided (cf. Bechtel & Richardson, 1993, p. 214). **R** also avoids circularity in another way; namely – again *contra* Fodor (1975) – by *not* equating representations with states [or products] of an internal symbol system. Given the nonsymbolic character of PDP, the status of distributed representations could not disinterestedly be evaluated by such an obviously prejudicial standard. Hence, whereas the classicist, "symbolic entity" notion of representation begs the issue at hand, my appeal to 'things' and a 'stands-for relation', however, does not.[8] Simply put, **R** is neutral not just among the kinds of things that can be representations, it is neutral as to their internal or external status as well. Just as important, **R** is neutral as to the processes – whether classicist or connectionist, natural or artificial, etc. – by

[4]The obviousness of the process-thing distinction is apparently lost on some connectionists (e.g., see Zhang & Norman, 1994, 87-94). As I shall show, this distinction bears directly on whether there even are any distributed representations.

[5]Don't read too much into 'existence', for it does not follow that pictures of, say, unicorns, couldn't be representations. The point, simply, is that for a representation to *be* a representation, it must be *about* something, and that something has to *be* in some sense, even if only as an uninstantiated concept.

[6]According to democratic theory, so too would Mona herself, if she had been elected to Congress. Admittedly, it would be odd to call her 'a representation'. We wouldn't have to, however, because we have a term that denotes *persons* in a stands-for relation; namely, 'representative'.

[7]That there are "witless" systems oblivious to the representations they either manipulate or produce (Haugeland, 1991, pp. 69-70) is a fact that in no way effects the ontic status of the representations themselves: If something meets **R**'s conditions, then regardless of the cognitive status of the system that produced it, regardless of whether anything is aware of it, and regardless of whether something can be said to have it, it is a legitimate representation.

[8]Hatfield (1988, 1991), Haugeland (1991), and Dretske (1988) each advocate similar general conceptions, conceptions *also* predicated on the classical – i.e., historical – notion of 'a representation' as an entity in a 'stands-for relation'. Nevertheless, there are significant differences among our accounts.

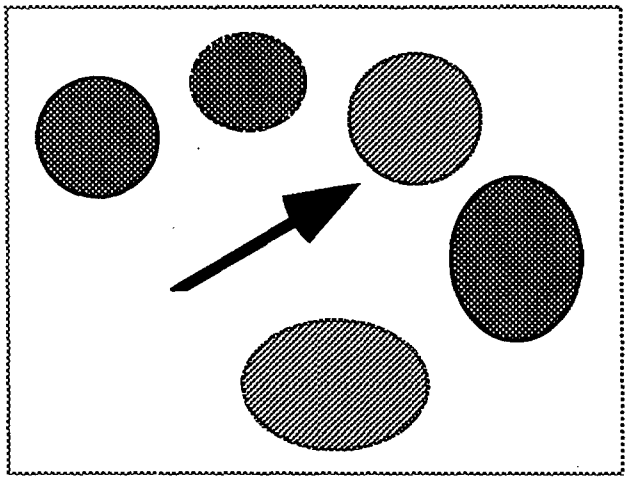


Fig. 4

which representations are produced.[9]

Still, because "almost anything may stand for almost anything else"[10] – and hence satisfy **R**'s conditions – it follows that almost anything can *be* a representation. For example [see **Figure 4**], assume that the shaded objects are real rocks, arranged on the ground in such a way to depict the location of my apartment building within the complex in which I live. If we stood over this arrangement of rocks, I would indicate my apartment building by saying "I live here," and then point to the rock indicated by the arrow. Because each of the rocks stands for a particular building in my apartment complex, according to **R**, each rock is a representation, as is the arrangement of rocks itself.[11]

The moral, therefore, is that anything meeting **R**'s theory-neutral identity conditions would be a legitimate representation. The question, however, is whether distributed patterns of activation can do so.

[9]My appeal to neutrality here is a paraphrase of a claim Haugeland (1991) makes in his analysis of representational schema.

[10]Goodman (1976). Also see Putnam (1988, appendix).

[11]Despite Fodor's claims to the contrary, rocks can be representations (see Fodor, 1987, p. xi). But rocks are not 'products' of a representation-producing process, like, say, photos or mental images. For this reason, 'thing' is more preferable than 'product' to identify the kinds of entities that can be representations.

4.4 Are distributed patterns of activation representations?

The place to look for distributed representations isn't among the output, the input, or generally in the network, but rather in the processing-specifically, "in the patterns of activity generated by the system of interconnected units" (Churchland & Sejnowski, 1989, p. 236).[12] Are such patterns really representations? Yes, or at least given the right sort of interpretation, they can be. But what they are not, however, are *internal* representations. Indeed, they can't be internal representations since PDP is necessarily nonsymbolic [analog] computation. Here's why.

4.4.1 Distributed patterns of activity are not products of PDP, they are PDP

To claim otherwise is to collapse the distinction between the *process* [mechanism] of distributed representation – PDP – and the *products* of PDP. This common confusion manifests itself in one of three ways: (1) when an iconic representation of vectored activation patterns among the hidden units is treated as though it were *itself* a distributed representation; (2) when one explicitly collapses the process-product distinction;[13] and (3) when one treats an instance of processing as though it were a 'thing' produced by the processing. Of the three ways one can confuse the *process* of distributed representation with its *products*, from an ontological standpoint, the third is the most serious, for it utterly trivializes what it means to be a representation.

For example, consider the following account of distributed representations in the brain:

[12]Also see Haugeland (1991, p. 84); cf. Schreter (1994, 95-98).

[13]For example:

> The basic principle of distributed representations is that the representational system of a distributed cognitive task can be considered as a set, with some members internal and some external. Internal representations are in the mind, as propositions, productions, schemas, mental images, *connectionist networks*, or other forms. External representations are in the world, as physical symbols . . . or as external rules, constraints, or relations embedded in physical configurations. (Zhang & Norman, 1994; my emphasis).

> The sensory surface . . . of the brain can be seen to contain 'representations' of stimulation from the environment. These representations are almost always distributed. For example, one particular 'rod' in the eye's retina can be activated [by] endlessly many light patterns falling on the retina. . . . This can be compared to, say, wanting to represent *a* by the vector [1,1,1,0,0,0,0,0], *b* by the vector [0,1,1,1,0,0,0,0] . . . etc. (Schreter, 1994, 95)

To make the confusion perspicuous, let me idealize the above account: Imagine yourself being scanned by positron emission tomography [PET] while your visual system processes a rather ordinary visual stimulus, say, a stop sign. Let us counterfactually assume that PET imaging techniques are so advanced, such that while the visual stimulus is processed, the PET researchers can identify: (a) each of the millions of discrete rods, cones, retinal ganglion cells, and cortical cells that are activated; and (b) the connection strengths between each activated neuron [which, roughly, would be a function of a neuron's firing frequency and the amount plus type of neurotransmitter released]. May we then point to this superbly refined neuronal activation pattern and say 'Lo, an internal representation'? No, for such patterns of activation are not 'things' produced by the processing, *they are the processing.* This also applies to activation patterns among the hidden units, *mutatis mutandis.* As such, distributed patterns of activation fail to meet **R**'s conditions. Strictly speaking, therefore, distributed patterns of activation are *not* internal representations.

But may one point to the above neuronal activation pattern and say 'Lo, a representation'? Yes; the same holds for activation patterns among the hidden units. This is so because anything can be treated *as if* it were a representation. Indeed, doing so toward hidden unit level activations has been one way connectionists have been able to conceptualize PDP dynamical processing. Here's the catch: there is a big difference conceptualizing and providing an explanation. And although both real and as-if representations can meet **R**'s conditions, there is a big difference between an *as-if* representation and a *real* one. Since all as-if representations are constructs [such as the above rocks, for instance], no as-if representation is ever a necessary component to a mechanistic explanation if the mechanism – process – being explained doesn't traffic in internal content-bearing entities.

Here's the moral: given the right sort of interpretation, distributed patterns of activation, like anything else, can *be* representations. Yet since it's the interpretational process that makes them representations, calling them 'internal representations' only compounds the confusion. Thus, again, distributed patterns of activation are not 'things' produced by the processing, they are the processing. As such, even in **R**'s generous sense, they are not internal representations. But if connectionists are using representation-talk in a nontrivial fashion, and 'internal distributed representations' are needed for explanations of PDP, then distributed patterns of activation ought to be internal representations. Such is not the case: 'distributed representa tions' are constructs – interpretations imposed on the processing – they are not internal products of the processing. While such constructs are useful at the descriptive level, at the explanatory level, however, they are not needed: recall, PDP networks are not functionally decomposable, for nothing less than the entire network figures in the production of their task-specific outputs. Therefore, given the very weak commitment to internal representations in explanations of PDP, appeals to 'internal distributed representations' can be dropped without any loss of explanatory power.

5 Conclusion

How can connectionists consistently maintain that PDP isn't rule-driven, cut-and-paste symbol manipulation [which is what gives PDP its biological plausibility] and yet appeal to internal distributed representations when explaining a network's behavior? There are two ways consistency can be assured without sacrificing PDP's biological plausibility: *revisionism* and *eliminativism.* Each solution exacts a price.

Most connectionists opt for revisionism – they treat PDP as a fundamentally different sort of representation-producing processing, and they treat distributed representations as a fundamentally different sort of content-bearing entity. Thus, if what it means to be a representation depends on one's theory, and representation and processing are "deeply intertwined" in PDP ne-

tworks (Clark & Toribio, 1994, 403), then each token distributed representation could be called 'a representation', and internal representation-laden explanations of PDP wouldn't be inconsistent with nonsymbolic processing. While this defense no doubt assures consistency, it does so, as I have shown, at the expense of trivializing what it means to be a representation.

The eliminativist solution begins with the recognition that while it *is* possible to give a computational [and hence representation-laden] *description* of any complex system – PDP networks included – descriptions are not explanations. Thus, if by ascribing distributed representations to PDP networks what connectionists are doing is merely treating states of PDP *as if they were representations*, then their appeals to internal distributed representations wouldn't be inconsistent with non-symbolic processing. But from an ontological standpoint – the standpoint that really matters when *explaining* how PDP or any other complex system does what it does – there is a big difference between treating something *as if* it were a representation and something actually *being* a representation. Since distributed patterns of activation are not really representations [but rather the processing], connectionists can without loss of the explanatory power of PDP abandon the practice of ascribing such 'representations' to PDP networks. In fact, they ought to abandon the practice since PDP networks are not decomposable into component distributed representations. As such, it makes no sense to localize the network's behavior to such "internal representations," not only because the cognitive task to be explained is an emergent phenomena of a non-decomposable complex system, but because distributed representations are *not* internal states of PDP. They are constructs, nothing more.

Thus, the eliminativist solution entails embracing a form of antirepresentationalism. Such is the price to be paid for *nontrivially* assuring the biological plausibility of PDP. Counterintuitive though this solution may be, I think it is the correct one. Indeed, if (1) representation-talk is to have any explanatory efficacy, and (2) the cognitive behavior to be explained is an emergent property of a complex, dynamic system, the eliminativist solution should be the correct one. I hope I have shown why. Therefore, if PDP-style explanations generalize to account for biological cognitive processing, given that the operative notion of 'representation' at work in neuroscience is mere causal covariation, then internal representations are not doing the work they are commonly thought to do. As such, the need for internal representations in explanations of cognition is – at best – minimal. Thus, I feel, much of the internal representation-talk common in naturalistic discourse about cognition can safely can go by the board.[14]

References

[1] Bechtel W. & Abrahamsen A. (1991) *Connectionism and the mind: An introduction to parallel processing in networks*, Cambridge, MA: Basil Blackwell.

[2] Bechtel W. & Richardson R. (1993) *Discovering complexity: Decomposition and localization as strategies in scientific research*, Princeton, NJ: Princeton University Press.

[3] Brooks R. (1991) Intelligence without representation. *Artificial Intelligence*, 47, 139-159.

[4] Chomsky N. (1957) *Syntactic structures*, The Hague: Mouton.

[5] Churchland P. S. & Sejnowski T. J. (1989) Neural representation and neural computation. In W. G. Lycan (Ed.), *Mind and cognition: A reader* (pp. 224-252), Cambridge, MA: Basil Blackwell, 1990.

[6] Clark A. (1989) *Microcognition*, Cambridge, MA: MIT Press.

[7] Clark A. & Toribio J. (1994) Doing without representing? *Synthese*, 101, 401-431.

[8] Cummins R. (1989) *Meaning and mental representation* Cambridge, MA: MIT Press.

[9] Cummins R. (1992) Conceptual role semantics and the explanatory role of content. *Philosophical Studies*, 65, 103-127.

[10] Darden L. (1991) *Theory change in science: Strategies from Mendelian genetics*, Oxford: Oxford University Press.

[14] I am indebted to Andy Clark, Bill Bechtel, and Mark Rollins for their comments and suggestions.

[11] Dretske F. (1988) *Explaining behavior: Reasons in a world of causes*, Cambridge, MA: MIT Press.

[12] Elman J. L. (1992) Grammatical structure and distributed representations. *Connectionism: Theory and practice* (pp. 138-194), New York, NY: Oxford University Press.

[13] Fodor J. A. (1975) *The language of thought*, Cambridge, MA: Harvard University Press.

[14] Fodor J. A. (1985) Precis of The modularity of mind. *The Behavioral and Brain Sciences*, 8, 1, 1-5.

[15] Fodor J. A. (1987) *Psychosemantics: The problem of meaning in the philosophy of mind*, Cambridge, MA: MIT Press.

[16] Fodor J. A. & Pylyshyn Z. W. (1988) Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28, 3-71.

[17] George A. (1989) How not to become confused about linguistics. In A. George (Ed.), *Reflections on Chomsky* (pp. 90-110), Cambridge, MA: Basil Blackwell.

[18] Goodman N. (1976) *Languages of art: An approach to a theory of symbols*, (2nd ed.), Indianapolis, IN: Hackett.

[19] Hanson N. R. (1958) *Patterns of discovery* Cambridge: Cambridge University Press.

[20] Hatfield G. (1988) Representation and content in some (actual) theories of perception. *Studies in History and Philosophy of Science*, 19, 175-214.

[21] Hatfield G. (1991) Representation in perception and cognition: Connectionist affordances. In W. Ramsey, S. P. Stich, & D. E. Rumelhart (Eds.), *Philosophy and connectionist theory* (pp. 163-195), Hillsdale, NJ: Lawerence Erlbaum Associates.

[22] Haugeland J. (1991) Representational genera. In W. Ramsey, S. P. Stich, & D. E. Rumelhart (Eds.), *Philosophy and connectionist theory* (pp. 61-89), Hillsdale, NJ: Lawerence Erlbaum Associates.

[23] Hempel C. G. (1966) *Philosophy of natural science*, Englewood Cliffs, NJ: Prentice-Hall.

[24] Hinton G. E., McClelland J. L. & Rumelhart D. E. (1986) Distributed representations. In D. E. Rumelhart, J. L. McClelland, & PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition*, Vol. 1: *Foundations* (pp. 77-109), Cambridge, MA: MIT Press.

[25] Kirsh D. (1991) Today the earwig, tomorrow man? *Artificial Intelligence*, 47, 161-184.

[26] Langley P., Simon H. A., Bradshaw G. L. & Zytkow J. M. (1987) *Scientific discovery: Computational explorations of the creative processes*, Cambridge, MA: MIT Press.

[27] Newell A. & Simon H. (1972) *Human problem solving*, Englewood Cliffs, NJ: Prentice Hall.

[28] Nickles T. (1980) Introductory essay: Scientific discovery and the future of science. In T. Nickles (Ed.), *Scientific discovery, logic, and rationality* (pp. 1-59), Dordrecht: D. Reidel Publishing Company.

[29] Peacocke C. (1989) When is a grammar psychologically real? In A. George (Ed.), *Reflections on Chomsky* (pp. 111-130), Cambridge, MA: Basil Blackwell.

[30] Pinker S., & Prince A. (1988) On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition*, 28, 73-193.

[31] Port R. & van Gelder T. (Eds.) (1995) *Mind as motion: Explorations in the dynamics of cognition*, Cambridge, MA: MIT Press.

[32] Putnam H. (1988) *Representation and reality*, Cambridge, MA: MIT Press.

[33] Rumelhart D. E., McClelland J. L. & PDP Research Group (Eds.) (1986) *Parallel distributed processing: Explorations in the microstructure of cognition*, Vol. 1: *Foundations*, Cambridge, MA: MIT Press.

[34] Schreter Z. (1994) Distributed and localist representations in the brain and in connectionist models (*Technical Report 289*): University of Queensland.

[35] Sellars W. (1980) Behaviorism, language, and meaning. *Pacific Philosophical Quarterly*, 61, 3-25.

[36] Stich S. P. & Warfield T. A. (Eds.) (1994) *Mental representation: A reader* Cambridge, MA: Basil Blackwell.

[37] Stufflebeam R. S. (1995) Are there distributed representations in distributed representation? *Proceedings of the 1995 Midwest Artificial Intelligence and Cognitive Science Society Conference*, 12-16.

[38] van Gelder T. (1991) What is the "D" in "PDP"? In W. Ramsey, S. P. Stich, & D. E. Rumelhart (Eds.), *Philosophy and connectionist theory* (pp. 33-59), Hillsdale, NJ: Lawerence Erlbaum Associates.

[39] van Gelder T. (1995) What might cognition be if not computation? *Journal of Philosophy*, XCI, 7, 345-381.

[40] Zhang J. & Norman D. A. (1994) Representations in distributed cognitive tasks. *Cognitive Science*, 18, 1, 87-122.

Is Consciousness a Computational Property?

Gilbert Caplain
ENPC-Cermics, La Courtine
F-93167, Noisy-le-Grand, Cedex, France
E-mail: caplain@cermics.enpc.fr

Keywords: consciousness, knowledge, belief, artificial intelligence

Edited by: Matjaž Gams
Received: May 15, 1995 **Revised:** October 19, 1995 **Accepted:** October 30, 1995

We will outline a proof that consciousness cannot be adequately described as a computational structure and(or) process. This proof makes use of a well-known, but paradoxical, ability of consciousness to reach ascertained knowledge (as opposed to mere belief) in some cases. Although such a result rules out "naive reductionism", it does not fully settle the reductionism vs dualism debate in favor of the latter, but merely leads to some kind of weak dualism.

1 Introduction

The recent developments in the field of Artificial Intelligence (AI) have revitalized some philosophical questions about the nature of consciousness and conscious knowledge. Some advocates of "strong AI" hold that, in a not too far-away future, it will be possible to design and program computers which will display all abilities commonly attributed to humans, including consciousness.

The aim of this paper is to show that this strong-AI paradigm leads to a paradox, in relation to the ability of consciousness to reach ascertained knowledge.

In Section 2, we recall the fundamental distinction between *knowledge* and *belief*, and show how the ability to make such a distinction is narrowly associated to *consciousness*.

In Sections 3 and 4, we outline a proof that consciousness cannot be adequately accounted for as a computational process or structure.

Such a result could be interpreted as leading to dualism. However, in Section 5, we show that such an interpretation is not necessarily justified, and that our result merely implies what could be termed as *weak dualism*.

For the sake of brevity, our argument is merely outlined here; some notions it involves call for further developments.

2 Knowledge and belief

A familiar property of human consciousness is its ability to reach "ascertained knowledge" as opposed to mere belief. In some cases, a human being is able to detect that some belief is not merely a belief, but indeed a piece of knowledge. This ability is sometimes termed as *"the ability to produce statements which are unassailably true"*. It may be of interest to keep in mind that such ascertained statements are most generally partial, approximative, subject to future improvements, as exemplified through the developments of modern science.

In what follows, the word "knowledge" will be used in its strong sense of "ascertained knowledge".

It is important to point out that, in this framework, *a piece of knowledge is not merely a belief which happens to be true.* The knowledge of a true statement by a conscious being does not involve merely the belief in this true statement; it also involves a state of consciousness, within this conscious being, which constitutes a valid guarantee that this statement is true. This is the difference between *true belief* and *justified true belief*, i.e. knowledge.

This ability to discriminate between knowledge and mere belief is denied by a philosophical conception – *absolute skepticism* – which states that we are not able to know anything for sure. As a

matter of fact, although this ability of knowledge is a basic tenet of scientific thought, it is *essentially impossible to derive a proof* of this ability, for the following reason: any such derivation would necessarily rely on previously acquired elements of knowledge identified as such, which would indeed presuppose what we would attempt to prove, leading to circular reasoning. In other words, absolute skepticism is *intrinsically irrefutable*: within the framework of absolute skepticism, it is impossible to prove anything at all.

In spite of the apparent impossibility to provide a satisfactory definition of *consciousness*, adequately matching all that we have in mind when we refer to "being conscious", we may mention at least an *element* of such a description, related to the ability to reach knowledge: *any conscious being, whose consciousness is active at some moment, is able to know something for sure at that moment: the fact that "there is conscious impression there".* In other words, having some conscious sensation at some moment entails the knowledge that, at least, there is that conscious sensation. Thus, some minimal ability for knowledge is inseparably attached to any conscious being. (It is fortunate for us humans that we are able to know some more than that.)

However limited it may be, the capacity of knowledge that any conscious being possesses is a paradoxical property in some sense: it entails a strange capacity of *self-checking*. In the body of our knowledge, there are necessarily truths which are "basic", or "primordial", in the sense that we rightfully consider them as *self-evident*, without having any clear idea of how we got to know them. Examples of such statements are: our own existence, the real existence of the world external to ourselves, the ability of our senses to provide us with some reflection of that external world.

The paradoxical character of that capacity of self-checking, or **reflexivity**, of the conscious mind is not new to philosophy, as evidenced, for instance, by the theme of the *brain in a vat*, an improved version of Descartes' *malicious genius*: *"What can warrant you that you indeed exist, that your perceptions of an outside world are not complete fantasies inoculated by some malicious genius, that you are not in fact a brain in a vat, nourished with nerve impulses computed and combined so as to make you falsely believe that you are a conscious human being living on some planet Earth, that you have existed for several years now, and that you are just reading an article about conscious knowledge... whereas the truth about you would be completely different ?"* Unless we fall into absolute skepticism, we are compelled to admit that strange property of reflexivity, that capacity of knowledge, however paradoxical it appears to us.

Many more things should be mentioned about belief, knowledge and reflexivity: let us just briefly outline a few aspects, useful for what follows.

- A piece of knowledge does not necessarily presuppose an elaborate linguistic representation: in the above example of the minimal knowledge *"there is conscious impression there"*, it is not assumed that this knowledge is consciously expressed through a language in the mind of that conscious being.

- Although the capacity of knowledge appears here as an attribute of consciousness, it is clear that *we are not conscious, at every moment, of everything we know for sure.* This leads us to the notion of *latent knowledge.* When we say that an information is present in our consciousness, as a piece of knowledge, i.e. with the "certainty label" which comes with it, we will not mean that this information is present to our awareness at the precise moment; we will mean that it is *recorded somewhere in our consciousness, as a piece of knowledge*, in a latent state, ready to come to our remembrance. More precisely, we will mean that there is a record in our consciousness of *either* this information itself, ready for remembrance; *or* another more general information of which the information being considered is a special case, an *instanciation*, straightforwardly obtainable on request.

- The property of reflexivity does not contradict the frequently observed phenomenon of *thorough unjustified belief,* i.e. error, although the paradoxical aspect of reflexivity is reinforced by that possibility of error, which is sometimes invoked in favor of absolute skepticism. Let us just mention that a conscious being is *in principle* able to detect whether one of its beliefs is justified or not as knowledge, but it may happen that, by *lack of attention*, the conscious being does not

realize that detection *in practice* at some precise moment and mistakes an unjustified belief for a piece of knowledge.

3 The cognitive separation principle

Can a computer or a robot be conscious ? We will try to explore that question in light of the above considerations.

We are interested in a widespread conception about consciousness and mental processes, which can be expressed as follows:

Reductionist conception of consciousness: *The human brain and mind is fully describable, in all its complex functions, as an* automaton*: a structured physical system the complex operations of which are fully describable as* computational *processes (in a broad sense): collecting, treating and recording pieces of information (and controlling the body consecutively). As a consequence, what we call* consciousness *is some part of this computational process.*

We will show that this reductionist conception is false, – that an *automaton* thus defined cannot be, by itself, conscious – by noticing that such a paradigm cannot account for the capacity of knowledge (as opposed to belief) we mentioned above.

First, we will introduce a preliminary observation; then, we will outline the proof in Section 4.

Let A be an automaton; let I be a *method of information recording* in this automaton. This "method" can consist of e.g. recording in a certain memory part, and/or through some coding, and/or with a labelling; or the union of several such methods of recording. Besides, this method I can evolve in time.

We will denote $\Sigma(A, I)$ the set of all informations recorded in A according to the method I – we will say, for brevity: "recorded through I". For any such method I, let $T(A, I)$ be the statement:

$$T(A, I) = \text{"All statements included in } \Sigma(A, I) \text{ are true"}.$$

We suppose that $T(A, I)$ is included in $\Sigma(A, I)$. It is immediately clear that such a recording does not imply that all informations recorded through I are indeed true: $\Sigma(A, I)$ may very well contain $T(A, I)$ together with false informations (in which case $T(A, I)$ is itself false). The property we have to point out here is slightly different from that obvious observation.

We wonder whether, and in which way, a conscious being E, considering the automaton A and the recording method I, may acquire the knowledge of $T(A, I)$, which implies the knowledge that all informations contained in $\Sigma(A, I)$ are true.

It is conceivable, in principle, that the structure of A and of the method I is such that only true informations can come to be recorded through I. It is even conceivable that E may acquire the knowledge of that property by examining A and I. However, such a verification will be obtained through the observation that there is a certain matching between the structure of A and I on one hand, and the field of reality the recorded informations deal with on the other hand. Therefore such a verification requires an observation of A, I *and* the domain of reality being considered. What is impossible, however, is that the inclusion of $T(A, I)$ in $\Sigma(A, I)$ be sufficient in itself to provide E with the valid guarantee that all informations in $\Sigma(A, I)$ are true (which is stated by $T(A, I)$). Hence the:

Cognitive separation principle: *Let A be an automaton, I a method of recording of informations in A, E a conscious being. We denote $\Sigma(A, I)$ the set of informations recorded in A through I and $T(A, I)$ the statement "All informations in $\Sigma(A, I)$ are true". The inclusion of $T(A, I)$ in $\Sigma(A, I)$ cannot be sufficient to validly guarantee to E that $T(A, I)$ is indeed true, i.e. that all informations in $\Sigma(A, I)$ are true.*

This principle expresses that any kind of information recording in an automaton cannot contain in itself a sufficient validation of these informations: in other words, these informations cannot be validly confirmed in a purely *internal* way. Between a field of the real world on one hand, and any information recording supposed to describe that field on the other hand, there exists some kind of *separation* such that the sole observation of these recordings cannot be sufficient to validly confirm them. That *cognitive separation* is the origin of the name we suggest for this principle.

We must emphasize a point, which is impor-

tant for AI : it is perfectly conceivable that some method of information recording in an automaton indeed "select" only true statements, and that we be able to confirm that (interesting) property. It is the *checking* of that property which cannot be purely internal to the structure considered.

4 Outline of the proof

Now we will show that an automaton cannot be conscious. We will use the cognitive separation principle, and the previously mentioned property that any conscious being is reflexive. We will show that there is a contradiction, for an automaton, between these two properties: cognitive separation and reflexivity.

Let us suppose an automaton E which would be conscious. Such an automaton would then be reflexive. Therefore, it would meet three conditions that we will derive now.

As a conscious, reflexive being, this automaton would have an ability for knowledge. The informations known for sure by this being at a certain moment would be the informations recorded through some method of recording, denoted C, corresponding to the *certainty label* we mentioned in Section 2. Hence our **condition 1**: *There is a method of recording C, in the automaton, such that the informations recorded in E through this method at a given moment (i.e. the set $\Sigma(E,C)$) are the informations it has knowledge of at that moment.* This method of recording C corresponds to the certainty for E.

The automaton knows for sure that any information it knows for sure is true. In other words, it knows that any information recorded through C – any information in $\Sigma(E,C)$ – is true. Now, according to Condition 1, the knowledge of an information by the automaton is equivalent to its inclusion in $\Sigma(E,C)$. Hence our **condition 2**: *Among the informations contained in $\Sigma(E,C)$, stands the information: "All informations in $\Sigma(E,C)$ – i.e. all informations E has knowledge of – are true" – a statement that we denoted $T(E,C)$.*

Our third condition states that the realization of Condition 2 would be sufficient to validly warrant to E that $T(E,C)$ is true. This is why. We could imagine, for a moment, that E makes sure of $T(E,C)$ – i.e. of the fact that the informations it knows for sure are true – by another means: by observing its own structure and inferring from this observation that any information in $\Sigma(E,C)$ – i.e. any information it is certain of – is true. But in this case, it would be necessary that the principles of inference and observation used by the automaton be validly warranted to it; then, according to Condition 1, these principles should be included in $\Sigma(E,C)$ and *their validity should be guaranteed* previously *to such a study!* In other words, *to infer $T(E,C)$, it would be necessary to already know $T(E,C)$!* Finally, the realization of Condition 2 would be the only means for the automaton to get such a guarantee. Hence our **condition 3**: *The realization of Condition 2, i.e. the recording of $T(E,C)$ through C, is sufficient for guaranteeing to E that $T(E,C)$ is true, i.e. that all informations it is certain of are true.*

Now, this Condition 3, a consequence of the hypothesis that our conscious automaton E is reflexive, is in contradiction with the cognitive separation principle, which applies to this case in the following way: *Let E be our conscious, hence reflexive, automaton; let C be the recording method we have considered, corresponding to the certainty for E. The recording of $T(E,C)$ through C cannot be sufficient to guarantee to any conscious being – hence, to E itself – that $T(E,C)$ is true.*

This contradiction implies that a conscious automaton could not be reflexive. However, we have mentioned that any conscious being is reflexive. Therefore, *an automaton cannot be conscious.* In other words, consciousness is not a computational phenomenon, in the broad sense outlined previously.

The derivation we have just presented, especially Condition 3 and its contradiction with the cognitive separation principle, highlights the paradox of knowledge we outlined above and reveals a rather remarkable property of consciousness: the *cognitive separation* between a field of reality and recorded informations supposed to describe it, does not extend to consciousness. In last resort, fundamentally, a conscious being builds its knowledge of reality only from conscious impressions, in the most general sense, i.e. from data "recorded" in its consciousness. However, paradoxically, this fact does not preclude the ability for (ascertained) knowledge, as would be the case if the conscious being were fully describable as an automaton.

5 Concluding remarks

The result we have outlined here could be interpreted as leading to dualism, together with challenging "strong AI". Things are not so clear-cut in fact.

First, this result does not necessarily preclude the possibility to make "conscious machines", for the following reason: we know a biological evolutive process – the development of our brain during gestation – which leads to the emergence of consciousness, whatever the nature of "consciousness" and the laws of that "emergence" will turn out to be. Therefore, it might be possible that some *other* evolutive processes of a more "artificial" kind – building and programming artificial automata – also lead to the emergence of consciousness. However, in this event, the consciousness thus emerging would not really be contained in the computational structure and process thus realized. Strictly speaking, the automaton would not be conscious, but a consciousness would have emerged in connection to it.

It would be tempting to state that, as a consequence of our result, "consciousness is not a physical phenomenon". In fact, in the absence of a sufficiently general definition of what we call a "physical phenomenon", such a statement is meaningless. More significantly, however, we can state that conscious phenomena cannot be *fully* accounted for by the principles of physics and cybernetics *known to date.* In this sense, our result leads to what can be termed as *weak dualism*, as opposed to a *strong dualism*, in the manner of Descartes for instance, which would posit a sharp separation between "mind" and "matter" and/or state that the realm of "mind" would not be amenable to scientific investigation.

Cracks in the Computational Foundations

Paul Schweizer
Centre for Cognitive Science
University of Edinburgh
Scotland
E-mail: paul@cogsci.ed.ac.uk

Keywords: computational paradigm, mental content, consciousness

Edited by: Matjaž Gams
Received: May 15, 1995 **Revised:** October 25, 1995 **Accepted:** November 8, 1995

The main thesis of the paper is that the computational paradigm can explain neither consciousness nor representational content, and hence cannot explain the mind as it standardly conceived. Computational procedures are not constitutive of mind, and thus cannot play the foundational role they are often ascribed in AI and cognitive science. However, it is possible that a computational description of the brain may provide a scientifically fruitful level of analysis which links consciousness and representational content with physical processes.

1 Introduction

Cognitive science and 'strong' AI are founded on the idea that computation is the theoretical key to the mind. Thus computation (of one sort or another) is seen as the appropriate scientific paradigm for understanding and explaining mental phenomena (e.g. Johnson-Laird, 1988). In turn, two features standardly held to be essential to the mind are intentionality and consciousness (Searle, 1992). Thus cognitive science and AI are, prima facie, committed to the view that computation is the theoretical key to explaining both intentional content and conscious experience. However, there are strong reasons for concluding that computation is singularly incapable of accounting for either of these essential features, which implies that computation is *not* an adequate foundation for a general theory of the mind.

In support of this view, I examine two basic consequences of the computational paradigm, and show how these consequences render intentionality and consciousness inexplicable within a purely formal framework. In the final section I consider the *possible*, though less fundamental role of a computational approach.

2 Mind/Program Identity and the Problem of Consciousness

The first consequence of the computational paradigm that I will examine is the identification of the mind with the level of the program or abstract procedure. It is held that the mind is to be characterized by algorithms and assorted formal architectures, and these can be instantiated in any number of *different* physical media. The computational pardigm produces a hierarchy of levels of description, where the most pronounced separation in levels is formed by this fundamental gap between hardware and software. Here, the salient distinction is that the software level is *abstract*, as evinced by its multiple realizability, while the hardware level is *mechanical*, and is concerned with the actual physical properties and behavior of a material device.

This distinction underlies the well known claim that the mind is to the brain as a program is to the electromechanical hardware of a computer (again, see Johnson-Laird, 1988). Accordingly, it is held that the mind does not reside at the level of the brain as a biological organ, and what is of interest to the science of mind is the *program*, not the bowl of porridge in which it happens to be implemented. So this identification of the mind with (some

high stratum within) the software level will be taken as one of the distinguishing characteristics of the computational paradigm, wherein the brain is dismissed as a contingent and uninteresting hardware system.

This distinguishing characteristic is then carried over to the attempted explanation of consciousness. Abstract computational role, rather than the medium of implementation, is held to be responsible for conscious mental states (Lycan 1995, Cole 1994). An immediate consequence of this view is that isomorphism of abstract structure is then sufficient to guarantee the identity of conscious states *across* implementations based in different physical media. For example, the qualitative aspect of the conscious visual presentation of the color blue is said to be determined by the particular *role* this type of quale plays in the computational structure of the human perceptual system. This system happens to be embodied in the particular neurophysical substrate that evolution has bestowed upon us as higher primates, but this particular brand of hardware has nothing essential to do with the nature of blue qualia. If this same computational structure were realized in a completely different type of physical medium, then this alternative realization would enjoy qualitatively identical experiences of blue.

However, such an approach cannot provide an adequate account of the causal basis of conscious experience. As noted above, computational processes are *abstract*, while conscious experiences are *not* abstract; they are actual, occurrent phenomena extended in time. Furthermore, they are by nature qualitative, while abstract structures are devoid of any qualitative dimension. Like numbers, sets and differential equations, systems of rule governed symbol manipulation (as well as connectionist networks formally conceived) are colorless and tasteless abstractions which exist neither in time nor in space, and which lack causal efficacy as well as qualitative aspect. Only through implementation in terms of specific material systems can such abstractions enjoy a presence in the actual world, and only through an instantiated presence can such formalisms have the power to produce any concrete effects.

So the computational approach makes a critical error by espousing multiple realizability as a hallmark of the theory, while simultaneously contending that qualitatively identical conscious states are preserved across different kinds of realization. The latter is the claim that a substantive *invariant* obtains over radically different physical systems, while the former is the claim that *no* internal physical regularities need be preserved. And this implies that there is no *actual* internal property which could serve as the basis for the invariant conscious phenomena. The computationalist cannot rejoin that it is formal role which supplies this basis, since formal role is abstract, and abstract features can only be *instantiated* via actual properties, but they cannot *produce* them.

Hence formal processing structure is ontologically the wrong kind of thing to produce real events extended in time and possessing non-abstract properties. The material brain must do the causal work of the mind, so if conscious states are real, then their ultimate cause must be the brain. In this manner, conscious experiences are properly seen as *hardware* states that realize an abstract computational role. This abstract role remains a software concern, while the *actual* properties of consciousness are a feature of the material substrate. So the same abstract role could be realized by a different material substrate which lacked the particular properties which distinguish the occurrent reality of qualia from the abstract role which they instantiate. And from this it follows that a different type of material system could realize the same information processing structure as the human mind and yet fail to be conscious. The mind is essentially *unlike* a computational formalism, since it cannot be divorced from its particular physical substrate, and this contravenes the hardware/software distinction and the attendant principle of multiple realizability.

3 Input/Output Boundaries and the Problem of Content

Intentionality is perhaps the most important feature standardly invoked to distinguish the mental from the non-mental. In the tradition of Brentano (1874), the *essence* of mental states is comprised by their 'aboutness' or 'directedness'. For example, the propositional attitude of believing that snow is white, is an exemplary case of a mental state that is directed towards something else as its object; the belief is *about* snow. Similarly, the per-

ceptual state of seeing a tree is *directed* towards a mind-independent object in the environment. And this ability to be 'about' things is a central distinguishing characteristic of mental states. In sharp contrast, trees, snow, electromagnetic radiation and other physical objects and events are not 'about' anything.

This traditional criterion of mentality is addressed in classical AI and cognitive science through the use of 'mental representations' as a central explanatory device (again, see Johnson-Laird, 1988). Mental representations are posited as the internal cognitive structures that encode the information utilized by intelligent systems. And according to the classical view, mental representations are the formal structures over which cognitive computations are preformed. In this manner, cognitive science and AI are able to posit structures that are manipulated formally, but which appear to possess a rich semantical dimension. As the terminology suggests, mental representations are supposed to be 'about' the associated objects and states of affairs, and hence serve as the theoretical basis for explaining the mind's intentional content. So mental representations are designed to ground the claim that the formal procedures in question are genuinely *meaningful*, and hence that the computational paradigm is able to account for this key feature of the mind.

However, I think that another critical error is made at this point. Computation is essentially a matter of transformations performed on uninterpreted syntax, so that formal structure *alone* is sufficient for all effective procedures. The specification and operation of such procedures makes no reference to the intended meaning of the symbols involved. Indeed, it is precisely this limitation to syntactic form that has enabled computation to emerge as a mathematically rigorous discipline (see, e.g., Boolos & Jeffery, 1989). But then the purported content of mental 'representations' is rendered superfluous to the algorithms that comprise the putative mental processes of cognitive science. The distinguishing criterion of mentality is lost, since the intended interpretation of the mental syntax makes no difference to its computational properties.

According to the Church-Turing thesis, every computable function is computed by some Turing machine. And every Turing machine is expressible as a finite table of instructions for manipulating the symbols '0' and '1', where the 'meaning' of the manipulated symbols is entirely ignored. There is nothing intrinsic to the formal machine that would indicate whether the syntactic transformations were computations of numerical functions, tests for the grammaticality of linguistic expressions, proofs of theorems in first-order logic, or answers to questions posed during a session of the Turing test. And many (negative) results in mathematical logic stem directly from this type of separability between formal syntax and intended meaning. The various upward and downward Löwenhein-Skolem theorems show that formal systems cannot capture intended meaning with respect to cardinality. And Gödel's incompleteness results involve taking a formal system designed to be 'about' the natural numbers, and systematically reinterpreting it in terms of its own syntax and proof structure. As a consequence of this 'unintended' interpretation, Gödel is able to prove that arithmetical truth (an exemplary *semantical* notion) cannot, in principle, be captured by finitary proof-theoretic means (again, see Boolos & Jeffery, 1989).

These very powerful results on the inherent limitations of syntactical methods would seem to cast a cold and sobering light on the project of explicating *mental content* in computational terms.[1] Indeed, they would seem to render hopeless such goals as providing a computational account of natural language semantics or propositional attitude states. Non-standard models exist even for such rigorous and strictly defined realms as formal arithmetic and fully axiomatized geometry. And if formal arithmetic cannot even impose isomorphism on its various models, how then can a program designed to process a particular natural language, say Chinese, supply a basis for the claim that the units of Chinese syntax possess a *unique* meaning?

The only viable strategy for solving this 'symbol grounding problem' is to make direct appeal to the actual environment in which the cognitive system is embedded, and the entire history of interactions between the two. However, such a strategy transgresses the limits imposed by a pu-

[1]However, I do not wish to advocate the view that Gödel's results alone establish that the human mind *cannot* be a finitary proof-theortic device.

rely computational theory of mind. If the mind is to be identified with a computational formalism, then the input/output specifications of the formalism define the boundaries of the mind. As indicated above, effective procedures, as such, are closed systems of syntactic manipulation, and the rules for such manipulations are defined only in terms of the formal structure of the input strings.

So it follows as a consequence of the computational paradigm that the mind cannot interact with the objects in the environment that its mental representations are about. Instead, these objects must impinge upon the mind's surface and be translated into the appropriate forms of input, and the mind then processes these input signals. But these signals are not the environmental objects themselves; rather they are certain effects these objects have on the system's sensory transducers. In turn, these sensory effects cannot uniquely determine the objects that caused them, and hence cannot distinguish between any number of different sources which produce the *same* input signals (e.g. they cannot distinguish sufficiently refined virtual environments from real ones).

The symbol grounding problem cannot be solved *within* the computational paradigm, and therefore formal procedures must *presuppose* content in order to give a cognitive dimension to syntactical manipulations. The purported content of mental representations can only be specified from outside the representational system, through primitive appeal to a variety of factors, including the immediate physical environment, the evolutionary history of the human organism, and the sociolinguist community. These factors are purely external to the input/output boundaries of the cognitive formalism, and semantic content itself cannot be recovered from the associated patterns of syntactic manipulation. Thus computation cannot account for mental content; instead, it must be *projected onto* these processes from outside, and hence content is assumed rather than explained by the theory.[2]

[2]Proponents of connectionism often maintain that learning episodes provide the link with the environment required by intrinsic semantics. But this claim is easily refuted by the observation that once the network has been trained and all the connection weights established, a duplicate system can then be constructed which will behave in exactly the same manner as the 'grounded' network, but which has had *no* prior contact with the environment.

4 The Possible Role of Computation

So, is computation simply a misleading metaphor, or can it still play an important role in the scientific elucidation of mind? Formal processes are not *constitutive* of mind, as claimed by orthodox computationalism (e.g. Newell & Simon 1976). Rather than providing an answer to conceptual or foundational questions such as 'what is the ultimate nature of a mental state?' computational analysis can perhaps serve the more modest goal of mathematically describing various instantiated systems that are presently known to exhibit genuine mental capacities. Many of the traditional disputes concerning the relation between computation and cognition have revolved around much stronger and more ideological claims, perhaps initiated by the provacative tone of Turing's (1950) original article, in which computationally mediated behavior is taken as the *hallmark* of mentality. And again in Newell and Simon's work, 'physical symbol systems' are stipulated as providing both necessary and sufficient conditions for intelligent behavior.

In contrast, I think these broadsweeping (and rather premature) claims should be disengaged from a properly scientific approach which aims in the reverse direction. Rather than attempting to identify or define intelligent systems *in advance*, the attribution of computational structure should first be explored as a potentially useful handle on given systems whose behavior is, for independent reasons, already deemed cognitively significant. In other words, we should start with uncontroversial cases of intelligence and see if computation can yield an interesting analysis of these known instances. This is a bottom-up empirical approach, rather than a top-down legislative one, and it could prove scientifically fruitful without commitment to any presuppositions about the 'ultimate nature' of cognition.

As a bottom-up, empirical research enterprise, cognitive science should be concerned with finding a description *of the brain* as an instantiated computational system. If the formal approach is to have any application, it must be tied to the brain as a physical machine; details of the conjectured abstract procedures are substantive empirical hypotheses, which, from the very beginning,

must be calibrated against neurophysiological reality. There is no justification for restricting the science of mind to the level of the program, since the hardware is already given by nature, and our task is to discover *whether* there is a useful description of this hardware under which it can be viewed as the realization of a formalism.

The pronounced theoretical attraction of formal systems is that they can, at least in principle, supply the missing link between high level intentional description and low level physical mechanisms. It is the fact that formal systems are *both* semantically interpretable and realizable in the material world that makes them important as a potential bridge between mental content and mechanical causation. The computational level cannot *define* the mind, since it can account for neither semantics nor consciousness. But internal algorithms can, in theory, serve as the intermediate level translating mental content into the causally efficacious level of neurophysiology, which in turn yields the overt behavioral manifestations of intelligence.

Advocates of the computational paradigm often claim that if a particular piece of behavior is to be understood in terms of the beliefs and desires of the agent (i.e. if we are interested in genuinely *mental* explanations), then neurophysiological considerations are irrelevant (Fodor 1978, Pylyshyn 1984). According to computationalists, it is explanation in terms of representational content which concerns the science of mind, and this content resides at the abstract rather than the mechanical level. However, as section 3 indicates, this appeal to intentional content supplies an equally good reason for rejecting *computational* description. Pure syntactical transformations are just as incapable of revealing intentional content as are the causal transformations governing brain processes. Appeal to the level of formal symbol manipulation (or connection weights among hidden units) adds no intrinsic explanatory insight, since computations *themselves* must be semantically interpreted in order to be informative. In the case of both neurophysiology and computation, the semantics must be projected onto these processes through purely external considerations.

Thus if one assumes the framework of intentional explanation as a starting point, then the addition of computation is theoretically idle, unless it provides an explicit *translation* of intentional content into physical states and mechanisms. On this picture, the potential scientific value of the computational paradigm would lie in providing a high level description of the physical substrate which unites the realm of abstract content with the realm of brain mechanics, thereby rendering semantic value causally potent. Such a result would discern formal regularity in the complex morass of neurophysiological activity, and would bridge the explanatory gap between mentalistic accounts and physical processes.

In addition, such an approach to the mind/brain could provide a unifying link between conscious experience and representational content. Conscious events are occurrent brain states which directly encode representational content, e.g. information about the environment in the case of conscious perceptual experience. Thus the cognitive manipulation of these physical states should comprise a direct computational link between semantics and brain mechanics. The computational structure realized in the brain must be such that the conscious states of the brain reflect the semantic interpretation of the formalism, and where the physical instantiation of the formalism governs the material transformations and interrelations between sentient states. This would provide a semantically interpretable account of the electrophysiological processes underlying consciousness, and would thereby yield a unified perspective on the mind/brain in terms of its two essential features.

Perhaps this type of computational project will seem overly ambitious. But I would contend that it is the *only* significant role that computation could play in an explanatory theory of mind. If it cannot provide an explicit link between interpretation and instantiation, then computation must be relegated to the marginal status of *modelling* cognitive phenomena, in precisely the same weak sense of simulation in which computation can be used to model meteorological or economic phenomena. There is no doubt interest and predictive value in such simulations, but they fall far short of being explanatory theories.

References

[1] Boolos G. & Jeffery R. (1989) *Computability*

and Logic. Cambridge University Press.

[2] Brentano F. (1874) *Psychology from an Empirical Standpoint.*

[3] Churchland P. (1989) *A Neurocomputational Perspective.* MIT Press.

[4] Cole D. (1994) Thought and Qualia. *Minds and Machines*, 4, p. 283-302.

[5] Dretske F. (1995) *Naturalising the Mind.* MIT Press.

[6] Flanagan O. (1992) *Consciousness Reconsidered.* MIT Press.

[7] Fodor J. (1978) *The Language of Thought.* Thomas Y. Crowell.

[8] Johnson-Laird P. (1988) *The Computer and the Mind.* Harvard University Press.

[9] Lycan W. (1995) *Consciousness.* MIT Press.

[10] Newell A. & Simon H. (1976) Computer Science as Empirical Inquiry: Symbols and Search *Communications of the Association for Computing Machinery*, 19, the Tenth Turing Lecture.

[11] Putnam H. (1988) *Representation and Reality.* MIT Press.

[12] Pylyshyn Z. (1984) *Computation and Cognition.* MIT Press.

[13] Schweizer P. (1994) Intentionality, Qualia and Mind/Brain Identity. *Minds and Machines*, 4, p. 259-282.

[14] Schweizer P. (1995) Physicalism, Functionalism and Conscious Thought. *Minds and Machines*, 5, in press.

[15] Searle J. (1992) *The Rediscovery of the Mind.* MIT Press.

[16] Turing A. (1950) Computing Machinery and Intelligence. *Mind*, 59, p. 433-460.

Gödel's Theorems for Minds and Computers

Damjan Bojadžiev
Institute "Jožef Stefan", Jamova 39, 61111 Ljubljana, Slovenia
Phone: +386 61 1773 768, Fax: +386 61 1258 058
E-mail: damjan.bojadziev@ijs.si, WWW: http://nl.ijs.si/~damjan/me.html

Keywords: Gödel's theorems, self-reference, artificial intelligence, reflexive sequences of theories

Edited by: Xindong Wu
Received: May 9, 1995 **Revised:** November 16, 1995 **Accepted:** 30 November, 1995

Formal self-reference in Gödel's theorems has various features in common with self-reference in minds and computers. These theorems do not imply that there can be no formal, computational models of the mind, but on the contrary, suggest the existence of such models within a conception of the mind as something that has its own limitations, similar to those which formal systems have. If reflexive theories do not themselves suffice as models of mind-like reflection, reflexive sequences of reflexive theories could be used.

1 Introduction

At first sight, the designation of the topic of this special issue, 'MIND <> COMPUTER', also transcribed as 'Mind NOT EQUAL Computer', looks like a piece of computer ideology, a line of some dogmatic code. But there are as yet no convincing artificial animals, much less androids, and computers are not yet ready for the unrestricted Turing test. Although they show a high degree of proficiency in some very specific tasks, computers are still far behind humans in their general cognitive abilities. Much more, and in much more technical detail, is known about computers than about humans and their minds. Thus, the required comparison between minds and computers does not even seem possible, much less capable of being stated in such a simple formula.

On the other hand, it could be argued that it is precisely because we do not know enough about ourselves and our minds that we can make comparisons with computers and try to design computational models. This is especially so because we also do not know exactly what computers are incapable of, although we have some abstract, general results about their limitations, such as Turing's theorem about the inability of an idealized computer to determine for itself whether its computation terminates or not. This theorem, and related results by Gödel and Church, are frequently used in arguments about the existence of formal models of the mind; interestingly enough, they have been used to argue both for and against that possibility. As a preliminary observation, it can be noted that the "negative" use of limitative theorems, as these meta-mathematical results are called, is less productive in the sense that the faculty by which mind is supposed to transcend "mere" computation remains essentially mysterious. The "positive" use of the theorems promotes a more definite, less exalted view of the mind as something which has its own limitations, similar to those which formal systems have. This paper argues for the latter view, exploring the common feature of all these theorems, namely self-reference, and focusing on Gödel's theorems.

2 Self-reference in Gödel's theorems

The application of Gödel's theorems to fields outside meta-mathematics, notably the philosophy of mind, was initiated by Gödel himself. He had a strong philosophical bent which also motivated his (meta)mathematical discoveries [31]. Gödel first thought that his theorems established the superiority of mind over machine ([31], [28:28-9]). Later, he came to a less decisive, conditional view: if machine can equal mind, the fact that it does cannot be proved [31]. This view also parallels

the logical form of Gödel's second theorem: if a formal system of a certain kind is consistent, the fact that it is cannot be proved within the system. Gödel's more famous first theorem says that if a formal system (of a certain kind) is consistent, a specific sentence of the system cannot be proved in it.

Gödel's theorems are actually special, self-referential consequences of the requirement of consistency: in a consistent system, something must remain unprovable. One unprovable statement is the statement of that very fact i.e. the statement that it is itself unprovable (first theorem): you cannot prove a sentence which says that it can't be proved (and remain consistent). Another unprovable statement in a consistent system is the statement of consistency itself (second theorem). In addition, if the formal system has a certain stronger form of consistency, the sentence which asserts its own unprovability, called the Gödel sentence, is also not refutable in the system. Rosser later constructed a more complicated sentence for which simple consistency is sufficient both for its unprovability and for its unrefutability. Similar sentences were constructed by others (e.g., Rogers and Jeroslow [4:65-6]), showing that consistent formal systems cannot prove many things about themselves. On the other hand, a formal system can retain all the insight into itself that is compatible with consistency: thus, although it cannot prove its Gödel sentence, if it is to remain consistent, it can prove that very fact, namely the fact that it cannot prove its Gödel sentence if it is consistent [22:114].

2.1 Implications of Gödel's theorems

The fact that a particular sentence is neither provable nor disprovable within a system only means that it is logically independent of the axioms: they are not strong enough to either establish or refute it - they don't say enough about it one way or the other. Saying more, by adding additional axioms (or rules of inference) might make the sentence provable. But in Gödel's cases, this does not work: even if Gödel's sentence is added as an additional axiom, the new system would contain another unprovable sentence, saying of itself that it is not provable in the new system. This form of self-perpetuating incompleteness might be called, following Hofstadter [10:468], essential incompleteness.

Gödel's theorems have uncovered a fundamental limitation of formalization, but they say that this limitation could be overcome only at the price of consistency; we might thus say that the limitation is so fundamental as to be no limitation at all. The theorems do not reveal any weakness or deficiency of formalization, but only show that the supposed ideal of formalization - proving all and only all true sentences - is self-contradictory and actually undesirable:

- what good is a formalization that can prove a sentence which says that it is *not* provable (first theorem)?
- what good is a formalization that can prove its consistency when it would follow that it is *not* consistent (second theorem)?

On the positive side, the theorems show that certain formal systems have a much more intricate, reflexive structure than formerly suspected, containing much of their own meta-theory.

Gödel's theorems show that the notions of truth and provability cannot coincide completely, which at first appears disturbing, since, as Quine says [21],

> we used to think that mathematical truth consisted in provability [p.17].

Gödel's theorems undermine the customary identification of truth with provability by connecting truth with *un*provability: the first theorem presents a case of

$$not\ provable \rightarrow true \qquad (1)$$

(if the sentence asserting its own unprovability is not provable, then it is true); the second theorem presents a case of

$$true \rightarrow not\ provable$$

(if the sentence asserting the consistency of the system is true, then it is not provable). However, the notion of truth has a problem of its own, namely the liar paradox, of which Gödel's sentence is a restatement in proof-theoretic terms. Thus, Gödel's theorems do not actually establish any disturbing discrepancy between provability and truth. Furthermore, the implication (1) above

is an oversimplification: assuming consistency, Gödel's sentence is not simply true, because it is not always true i.e. not in all interpretations (else it would be provable, by the completeness theorem, also proved by Gödel: provability is truth in *all* interpretations). The first theorem shows that if the system is consistent, it can be consistently extended with the *negation* of the Gödel sentence, which means that the sentence is actually false in some models of the system. Intuitively, without going into details, this could be explained by saying that in those models the Gödel sentence acquires a certain stronger sense of unprovability which those models do not support [1:391]. Gödel's theorem thus shows that there must always exist such unusual, unintended interpretations of the system: as Henkin says, quoted by Turquette [26]:

> We tend to reinterpret Gödel's incompleteness result as asserting not primarily a limitation on our ability to *prove* but rather on our ability to specify what we *mean* ... when we use a symbolic system in accordance with recursive rules.

Similarly, Polanyi says, though only in connection with the second theorem [19]:

> we never know altogether what our axioms mean [p.259]. We must commit ourselves to the risk of talking complete nonsense if we are to say anything at all within any such system [p.94].

This characterization of formal language sounds more like something that might be said about ordinary, natural language. Thus, if we take as a characteristic of ordinary language its peculiar inexhaustibility and the frequent discrepancy between intended and expressed meaning ("we never know altogether what our sentences mean; we must risk talking nonsense if we are to say anything at all"), Gödel's theorems would show that, in this respect, some formal languages are not so far removed from natural ones. Certain similarities between the self-reference in natural language and in Gödel's sentence and theorems have also been noticed at the lexical and pragmatic level (indexicals [25], performatives [10:709]). This line of thought, namely that the self-reference which leads to Gödel's theorems makes a formal system more human, so to speak, will be followed here to the conclusion that such systems are indeed suitable for modelling the mind.

2.2 Non-implications of Gödel's theorems

Some authors, especially those who attempt to apply Gödel's theorems to disciplines other than meta-mathematics, are handicapped by a more or less severe misunderstanding of the theorems. For example, Watzlawick, Beavin and Jackson state [29]:

> Gödel was able to show that it is possible to construct a sentence G which
>
> 1. is provable from the premises and axioms of the system, but which
> 2. proclaims of itself to be unprovable.
>
> This means that if G be provable in the system, its unprovability (which is what it says of itself) would also be provable. But if both provability and unprovability can be derived from the axioms of the system, and the axioms themselves are consistent (which is part of Gödel's proof), then G is undecidable in terms of the system [p.269].

Of course, this is completely garbled, but the authors nevertheless have very interesting ideas about applications of Gödel's theorems.

A less serious but more common misunderstanding is to overlook or tacitly drop the consistency premise in the first Gödel theorem. It is frequently stated that the theorem establishes the existence of a true sentence which is not provable. The theorem says that if the system is consistent, the sentence asserting its own unprovability is not provable. It might then seem that the sentence must be true, since that is what it says of itself; so, there is a true but unprovable sentence. But the theorem only says that the sentence is unprovable *if* the system is consistent; so, the sentence will likewise be true (in the intended interpretation) *if* the system is consistent. The difference is that between conditional and unconditional truth, and it is considerable, because the condition is

consistency of the system, and the second Gödel theorem shows that there is a problem with establishing that.

3 Formal models of the mind

Gödel's (first) incompleteness theorem can be expressed in the form: a sufficiently expressive formal system cannot be both consistent and complete. With this form, the attempt to use such formal systems as models of the mind invites the following brush off:

> Since human beings are neither complete nor consistent, proving that computers can't be both doesn't really help [R. Jones in *sci.logic*, May 1995].

A different intuition was followed by Wandschneider: the limitations of formalization revealed by Gödel's theorems prevent the use of formal systems as models of the mind [27]. Most authors, however, accept the comparison between mind and formal systems of the kind considered by Gödel, but reach different conclusions. For example, according to Haugeland [9],

> most people are agreed ... that [Gödel's] result does not make any difference to cognitive science [p.23].

According to Kirk [12], arguments against mechanisms based on Gödel's theorems are agreed to be mistaken, though for different reasons; cf. Dennett [6] and especially Webb [30]. These arguments try to establish the superiority of mind by suggesting that mind can reach conclusions which a formal system cannot, such as Gödel's sentence.

3.1 The basic incompleteness argument

Arguments about the relative cognitive strength of minds and machines usually invoke only the first Gödel theorem, although the second theorem also establishes the existence of a sentence which, if true, is not provable. The comparative neglect of the second theorem seems strange in view of the way in which the second theorem bears on applications of the first: establishing the existence of a sentence which is true (in the intended interpretation) but is not provable presupposes that the consistency of the system is already established. On the other hand, it might be expected that neglect of the second theorem would go hand in hand with misinterpretation of the first one as saying simply that there is a true but unprovable sentence. This frequently happens in the basic version of the argument from incompleteness: since any formal system (of a certain kind) contains a true but unprovable sentence, mind transcends formalism because mind can "see" that the unprovable sentence is true. This conviction can be traced, in various forms, from Penrose [17], [18] through Lucas [14] back to Nagel and Newman [16:100-1]. For example, Lucas [14] says:

> However complicated a machine we construct, it will ... correspond to a formal system, which in turn will be liable to the Gödel procedure for finding a formula unprovable-in-that-system. This formula the machine will be unable to produce as being true, although a mind can see it is true. And so the machine will still not be an adequate model of the mind.

The consistency premise is not very prominent here, but some suspicious phrasing is: 'producing as being true', 'seeing to be true', instead of the simpler and more to the point 'proving'. This way of comparing cognitive strength in humans and machines leaves out an obvious symmetry while emphasizing a dubious asymmetry. The symmetry is that, just as a formal system cannot prove a sentence asserting its own unprovability, unless it is inconsistent, so can a mind not do so, if it is consistent. The doubtful asymmetry between mind and machine concerns their possession of the notion of truth. The mind is supposed to have this notion in addition to the notion of provability, and is supposed to have no problems with it (but it does, namely the liar paradox). On the other hand, the machine is only supposed to be able to prove things (as its only means of establishing truth) without having, and apparently without being able to have, an additional notion of truth. But this is not so: for expressing the truth of the Gödel sentence (as opposed to proving it), even the most restricted definition of the truth predicate $true_1(x)$, covering sentences containing at most one quantifier, is sufficient [30:197].

3.2 Mind over machine $\omega : 1$?

A more intricate version of the argument from incompleteness considers adding a "Gödelizing operator" to the system. This form of the incompleteness argument was also advanced by Lucas [14]:

> The procedure whereby the Gödel formula is constructed is a standard procedure ... then a machine should be able to be programmed to carry it out too ... This would correspond to having a system with an additional rule of inference which allowed one to add, as a theorem, the Gödel formula of the rest of the formal system, and then the Gödel formula of this new, strengthened, formal system, and so on ... We might expect a mind, faced with a machine that possessed a Gödelizing operator, to take this into account, and out-Gödel the new machine, Gödelizing operator and all.

The sound part of this argument is already contained in the notion of essential incompleteness: a Gödel operator only fills a deductive "lack" of the system by creating a new one. Adding the Gödel sentence of a system as a new axiom extends the notion of provability and thereby sets the stage for a new Gödel sentence, and so on. Thus, a Gödel operator only shifts the original "lack" of the system through a series of displacements, without ever completing the system.

The Lucas argument, especially in the form advanced by Penrose [18], now centers on how far into the transfinite can a Gödel operator follow the mind's ability to produce the Gödel sentence of any system in the sequence

$$S_0, S_1 = S_0 + G(S_0), S_2 = S_1 + G(S_1), \ldots$$

$$S_\omega, S_{\omega+1} = S_\omega + G(S_\omega), \ldots$$

.....

A relevant result here is the Church-Kleene theorem which says that there is no recursive way of naming the constructive ordinals [10:476]. This would mean that a Gödel operator could only follow the mind's ability to produce Gödel sentences through the recursively nameable infinite [18:114]. Feferman's results on recursive progressions of axiomatic theories show that this is no real limitation, so that, as Webb [30] says,

> there is not the slightest reason to suppose that ... a machine could not model the 'ingenuity' displayed by a mind in getting as far as it can [p.173].

But for the purposes of this paper it is more interesting to observe that it does not seem plausible that the argument about the formalizability of mind should be decided by the outcome of the race between mind and machine over remote reaches of transfinite ordinality. And even if it makes sense to conceive of mind as always being able to out-reflect a reflective formal model, it would seem that the ability to perform the self-reflection is more important than the question of how far does this ability (have to) reach.

3.3 Reflexive sequences of reflexive theories

A further possibility in the direction of making reflexive formal models is to make the progression of reflexive theories itself reflexive. The usual ways of extending a reflexive theory by adding its Gödel sentence, or the statement of consistency (Turing), or other reflection principles (Feferman) are themselves not reflexive: what is added to a theory only says something about that theory, and nothing about the one which its addition produces. Thus, what is usually added to a theory does not anticipate the effect of that very addition, which is to shift the incompleteness of the original theory to the extended one. Of course, certain things about the extended theory cannot be consistently stated; for example, the sentence stating that its addition to a theory produces a consistent theory would lead to contradiction, by the second Gödel theorem. But the sentence which is added to a theory could make some other, weaker statement about the theory which its addition produces. If the procedure of theory extension operated not only on the theory it is to extend but also on a representation of itself, it could build on its own action and improve its effects. It could thus produce in a single step an extension which is much further down the basic sequence of extensions, produced by linear additions of Gödel sentences; the size of this ordinal jump could then be taken as a measure of the reflexivity of the procedure. This kind of procedure, operating on something which contains a

representation of that procedure itself, is already familiar from the construction of the Gödel sentence: the process of diagonalization operates on a formula containing a representation of that very process, and constructs a sentence which refers to itself as the result of the diagonalization which produced it ([10:446], [3]). Another example of a reflexive procedure of this kind would be the Prolog meta-circular interpreter, which can execute itself, though only to produce statements of iterated provability [5:536].

4 Self-reference in computers

In saying of itself that it is not provable, the Gödel sentence combines three elements: the representation of provability, self-reference and negation. In computer science, self-reference is more productive in a positive form, and in programs, programming systems and languages more than in individual sentences. The first ingredient in Gödel's sentence, the representation of provability, corresponds to the explicit definition of the provability predicate of a logic programming language in that same language. In the simplest case, specifying Prolog provability in Prolog, the definition consists of just a few clauses [5:536], comparable to those which express the conditions on the provability predicate under which Gödel's theorems apply. This definition of Prolog provability is then used as a meta-circular interpreter to extend the deductive power of the basic interpreter, for example by detecting loops in its proof attempts. This use of the meta-circular interpreter could be compared to the work of the Gödel operator on extending the basic, incomplete theory. Meta-circular interpretation is also applicable to other programming languages, notably LISP [23].

Generalizing meta-circular interpretation, provability can be specified in a separate metalanguage, and reflection principles defined for relating and mixing proofs in both languages. Such meta-level architectures [32] can be used to implement reflective or introspective systems, which also include an internal representation of themselves and can use it to shift from normal computation about a domain to computation about themselves [15] in order to achieve greater flexibility. Meta-level architectures are useful for knowledge representation, allowing the expression and use of meta-knowledge, and opening the possibility of computational treatment of introspection and self-consciousness [7:128]. For example, Perry suggested an architecture of self-knowledge and self in which indexicals mediate between bottom level representations, in which the organism is not itself represented, and higher levels at which it is represented generically, as any other individual [20].

5 Self-reference in minds

The basic lesson of Gödel theorems, namely that the ability for self-reflection has certain limits, imposed by consistency, does not seem to be less true of minds than it is of formal systems. Applied to minds, it would translate to some principled limitation of the reflexive cognitive abilities of the subject: certain truths about oneself must remain unrecognized if the self-image is to remain consistent [10:696]. This formulation recalls the old philosophical imperative which admonishes the subject to know himself. If this were simple or possible to do completely, there would be no point to it; the same goes for the explicit interrogative forms: who am I, where am I going, what do I want, ... Hofstadter [10] rhetorically asks:

> Are there highly repetitious situations which occur in our lives time and time again, and which we handle in the identical stupid way each time, because we don't have enough of an overview to perceive their sameness? [p.614].

Such an overview can be hard to achieve, especially in regard to oneself, as Laing's knots in which minds get entangled show [13]. In a similar vein, Watzlavick, Beavin and Jackson suggest that the limitative theorems show the mathematical form of the pragmatic paradoxes to which humans are susceptible in communication [29:221].

It may be that, as Webb says, the phrase 'the Gödel sentence of a man' is an implausible construction [30:x], but certain interpretations might be imagined, such as self-falsifying beliefs. On a humorous note, the Gödel sentence for a human could work like a recipe for self-destruction, activated in the process of its comprehension or articulation ("self-convulsive", "self-asphyxiative", "self-ignitive", ...). A more elaborate interpretation, as the paralysing effect

of some self-referential cognitive structure, is presented in Cherniak's story [11:269]. The history of logic itself records lethal cases (Philetas) and cases of multiple hospitalization (Cantor, Gödel). Of course, this is all anecdotal, speculative and inconclusive, but it does suggest that the apparent gap between minds and machines could be bridged, in two related ways:

- the vulnerability of minds to paradoxes of self-reference
- the implementation of self-referential structures in machines

The mind-machine gap could thus be reduced by emphasizing the formal, machine-like aspects of the mind and/or by building mindlike machines.

Finally, taking speculation one literal step further, the self-reference in Gödel's sentence can be compared to a formal way of self-recognition in the mirror, by noticing the parallelism between things (posture, gesture, movement) and their mirror images. The basis for this comparison is the way the Gödel code functions as a numerical mirror in which sentences can refer to, "see" themselves or other sentences "through" their Gödel numbers. The comparison, developed in [2], covers the stages of construction of Gödel's sentence and relates them to the irreflexivity of vision and the ways of overcoming it. The comparison attempts to turn arithmetical self-reference into an idealized formal model of self-recognition and the conception(s) of self based on that capacity. The motivation for this is the cognitive significance of the capacity for self-recognition, in mirrors and otherwise. The ability to recognize the mirror image, present in various degrees in higher primates and human infants, has been proposed as an objective test of self-awareness [8:493]. Self-recognition in the mirror is a basic, even paradigmatic case of self-recognition, the general case being the recognition of effects on the environment of our own presence in it. Self-recognition in this wider sense is the common theme of Dennett's conditions for ascribing and having a self-concept and consciousness [11:267]. Self-recognition is also the common theme of the self-referential mechanisms which, according to Smith [24], constitute the self:

- indexicality (self-relativity of representations)
- autonomy (recognizing one's own name)
- introspection (recognizing one's own internal structure)
- reflection (recognizing one's place in the world)

The comparison between formal and specular self-reference and self-recognition might also connect these contemporary attempts to base the formation of a self(-concept) on the capacity for self-recognition with the long philosophical tradition of thinking about the subject in optical terms.

6 Conclusion

It is not possible to see oneself completely, in the literal, metaphorical ("see=understand"), formal and computational sense of the word. Gödel's theorems do not prevent the construction of formal models of the mind, but support the conception of mind (self, consciousness) as something which has a special relation to itself, marked by specific limitations.

Acknowledgement

I am grateful to the editor, Xindong Wu, for his work on this paper, to Matjaž Gams, for encouraging me to write it, and to its referees, especially the one who noticed the remarkable claim that mathematics is useless because it fails to prove itself inconsistent.

References

[1] Bojadžiev, D., Sloman's view of Gödel's sentence, Artificial Intelligence 74 (1995) pp.389-93

[2] ———, Specular Self-Reference, in E. Gal, M. Marcelli, P. Michalovič (ed.), *Science and Philosophy in Shaping Modern European Culture III*, Bratislava 1995

[3] ———, Reconstructing Diagonalization(s), Yearbook of the Kurt Gödel Society 1989, Vienna 1990

[4] Boolos, G., *The Unprovability of Consistency - An Essay in Modal Logic*, Cambridge Univ. Press 1979

[5] Bratko, I., *Prolog programming for AI*, Addison Wesley 1990

[6] Dennett, D.C., On Alleged Refutations of Mechanism Using Gödel's Incompleteness Results, J. Phil. Vol. LXIX, No. 17, Sept. 1972

[7] Giunchiglia, F., Smaill, A., Reflection in constructive and non-constructive automated reasoning, in H. Abramson, M.H. Rogers (ed), *Meta-Programming in logic programming*, MIT Press 1989

[8] Gregory, R. (ed.), *The Oxford Companion to the Mind*, Oxford University Press, 1987

[9] Haugeland, J., *Mind Design*, Bradford Books 1981

[10] Hofstadter, D.R., *Gödel, Escher, Bach: An Eternal Golden Braid*, Basic Books 1979

[11] ————, Dennett, D.C., *The Mind's I - Fantasies and Reflections on Self and Soul* (1981), Penguin 1982

[12] Kirk, R., Mental Machinery and Gödel, Synthese 66, 1986

[13] Laing, R.D., *Knots*, Tavistock Publ. 1970

[14] Lucas, J.R., Minds, Machines and Gödel, Philosophy 36, 1961

[15] Maes, P., Nardi, D. (ed.), *Meta-level Architectures and Reflection*, North-Holland 1988

[16] Nagel, E., Newman, J.R., *Gödel's Proof*, NY Univ. Press 1958

[17] Penrose, R., *The Emperor's New Mind*, Oxford Univ. Press 1989

[18] ————, *Shadows of the Mind*, Oxford Univ. Press 1994

[19] Polanyi, M., *Personal Knowledge*, Routledge & Kegan Paul 1958

[20] Perry, J., Self-knowledge and Self-representation, in *Proceedings of the 9th IJCAI*, Vol.2, Los Angeles 1985

[21] Quine, W.v.Orman, The Ways of Paradox, in *Ways of Paradox and Other Essays*, Harvard Univ. Press 1966, 1977

[22] Robbin, J.W., *Mathematical Logic - A First Course*, W.A. Benjamin 1969

[23] Smith, B., Reflection and semantics in a procedural language, MIT TR-272, 1982

[24] ————, Varieties of Self-reference, in J.Y. Halpern (ed.), *Theoretical Aspects of Reasoning about Knowledge*, Proceedings of the 1986 Conference, Morgan Kaufmann 1986

[25] Smullyan, R.M., Chameleonic Languages, Synthese 60 (1984), pp.201-224

[26] Turquette, A.R., Gödel and the Synthetic A Priori, J. Phil. No. 57, 1950

[27] Wandschneider, D., Zur Eliminierung des Gödelschen Unvollständigkeitsproblems im Zusammenhang mit dem Antinomienproblem, Zeitschrift f. allg. Wissenschaftstheorie, 6:1, 1975

[28] Wang, H., Mind, Brain, Machine, Yearbook of the Kurt Gödel Society 1989, Wien 1990

[29] Watzlawick, P., Beavin, J.H., Jackson, D.J., *The Pragmatics of Human Communication*, W.W. Norton & Co., 1967

[30] Webb, J.D., *Mechanism, Mentalism, and Metamathematics - An Essay on Finitism*, D. Reidel Publ. Co. 1980

[31] Weibel, P., Schimanovich, W., Kurt Gödel: Ein Mathematischer Mythos (screenplay), ORF TV film, 1986

[32] Yonezawa, A., Smith, B. (ed)., Reflection and Meta- level Architectures, *Proc. Int. Workshop on New Models for Software Architecture*, Tokyo 1992

On the Computational Model of the Mind

Mario Radovan
FET - Pula, University of Rijeka,
Preradoviceva 1/1, 52000 Pula, Croatia
Phone: +385 52 23455 Fax: +385 52 212 034
E-mail: Mario.Radovan@efpu.hr

Keywords: mind, consciousness, computability, functionalism, language of thought, metaphor, hardware independence, connectionism

Edited by: Xindong Wu
Received: May 3, 1995 **Revised:** November 3, 1995 **Accepted:** November 7, 1995

The paper examines the power and limitations of the computational model of the mind. It is argued that conscious mind and human brain are not programmable machines, but that there are pragmatical reasons to assign them a computational interpretation. In this context, I speculate on the possibility that programmable machines exceed natural mind (in all kinds of mental abilities), but I also show that not all features of actual computer systems can be successfully mapped on the human mind/brain.

1 Introduction

The question of the relation between human mind and computational machines does not concern simply some observer independent phenomena in the objective world, but it primarily concerns our *attitude* toward these phenomena. In other words, this question cannot be answered/decided simply on the basis of empirical investigations of mind and/or machines, but primarily on the base of our *pragmatic needs* in the context of our efforts to understand and describe the phenomenon of *conscious mental states*, and to develop *powerful tools* which we hold useful for our survival and entertainment. Taken literally, mind is not machine, just as bird is not aeroplane (and even the less is aeroplane a bird). However, I hold that the assertion 'Mind is computer' should be intended primarily in the *figurative* sense; consequently, judgments on its validity should not be concerned so much with its literal truth as with the suitability of the scientific paradigm installed by such an assertion taken as a *metaphorical figure*.

There are a few positions concerning the relation (literal and metaphorical) between conscious human mind and computational machines. In this paper I tried to put forward the strongest reasons for, and the main weakness of, the following four positions:

1. Mind is *more* than machine could be. Machines are defined in *functional/syntactic* terms, while *conscious* mental states cannot be completely described in such terms, and even less could they be replicated by merely syntactically defined machine processes.

2. Mind is *less* than machine could be. There are theoretical results which open the possibilities of the development of such computational machines which shall far exceed the humble human mental abilities. And there is no reason to doubt that these possibilities will be realized.

3. As you *like* it. Human mind is a product of some biological processes which take part in the brain, and which are no more computational than those in the liver are. However, if there are pragmatical reasons to assign the computational interpretation to mind/brain processes, we can do it; but we should not mix the reality with the interpretative model.

4. When you *need* it. There is virtually no author in the scope of cognitive science who hasn't used the word 'mystery' when speaking on the conscious mind. Literal speech cannot express essential features of mysterious things; therefore, when concerned with the conscious mind, we are

often constrained to use metaphorical speech as the only possible tool of thought.

2 Mind Is More Than Machine

When we speak of the *mental* we intend the *conscious* mental state. Namely, conscious experience is "a necessary condition for the attribution of mentality" [13, p. 273], and "we have no notion of the mental apart from our notion of consciousness" [15, p. 18].

2.1 Mental States

A conscious mental state is a *property* of the substantial structure brain. We distinguish between *primary* and *secondary* properties; primary properties are defined as those that are *independent* of the subject (observer), while secondary properties are said to be *relational* in the sense that although causally dependent of the primary properties, they exist only *for* the subject (observer). In this context, a conscious mental state cannot be coherently conceived of as a secondary property because it has nothing to be related to; consequently, it should be conceived of as a primary property of the material structure brain. Searle says that "consciousness is a higher-level or *emergent* property of the brain" [15, p. 14]; he doesn't give an explicit definition of "higher-level property", but from the analogies he uses, these properties should be the primary properties.

The computational model of the mind says that mind is a kind of software while brain is a kind of hardware. As Searle put it, the "slogan" one often sees runs: "The mind is to the brain as the program is to the hardware" [15, p. 200]. In this context, the claim that a conscious mind is more than any programmable machine could ever become, can be defended on at least two grounds.

2.2 Emergent vs Imposed

Mental states are *emergent* primary properties of the material structure brain, while software could hardly be conceived of as a property of hardware, and in no way as an emergent property. Software is *imposed* on the hardware (from the outside), and there is not much sense in comparing conscious mind with software: consequently, the computational metaphor of the mind is simply not suitable. In other words, programming cannot be the right way to conscious mental states.

However, there are objections to such fast elimination of the computational paradigm of the mind; for example, it has been argued that some kind of software *could induce* (make to emerge) some kind of mental states on some kind of hardware! In principle, it could. Namely, programs (when loaded/active) have direct impact on the primary properties of the hardware: therefore, they could, in principle, induce all sorts of states. But it would be more than a miracle if any kind of software (as actually conceived) would ever induce a mental state on any type of hardware (as actually conceived). Furthermore, it is actually not possible even to work on the development of such kind of software because we currently don't know how mental states emerge in the human brain. And if it is not possible to purposely work on the development of such software, it is not wise to expect that such (mental) states could simply happen/emerge.

However, our actual ignorance concerning the nature and the ways of emergence of mental states is not the main problem here; namely, if we would know everything concerning the human brain/mind, we would very probably know also the fact that it is not possible to replicate mental states by mere computer programs. Indeed, there is no more reason to expect that the emergence of mental states could be caused by computer programs than there is to expect that the growth of grass could be: both events are natural phenomena, and programs can only *simulate* such phenomena, but not also *replicate* them. Hence, conscious human mind is more than any programable machine could ever become.

2.3 Mental vs Functional

There is an unbridgeable gap between the functional and the mental; functional properties can be described (and replicated) by formal (inanimate) systems, while mental states are intrinsically *first-person* and they cannot be completely described, but can be only *experienced.* Machines can simulate functional properties of the human mind, but not the mental ones: they can perform various well defined functions, but not also have/experience mental states. In other words, programmable machines can have *intel-*

ligence (defined as behavioral disposition), but not *mentality* (as an intrinsically first-person property). And mere intelligent behaviour is not enough for possessing/creation of mental states; moreover, "the relation of mental states to behaviour is purely contingent" [15, p. 23].

However, there are claims that conscious mental states could be reproduced by artificial means. For example, Dennett, who pleads for a "version of functionalism", says: "If all the control functions of a human wine taster's brain can be reproduced in silicon chips, the enjoyment will *ipso facto* be reproduced as well" [7, p. 31]. Perhaps; for if you duplicate all causes, you should duplicate also all the effects. However, functionalism is *ipso facto* not an approach which could lead us to the *real* (physical) "control functions" (whatever it means) of the brain.

There are a few "versions" of functionalism; let us try to collect the essence of the functionalist approach in the following statements. According to functionalism, any mental state can be defined in terms of the (1) sensory inputs, (2) causal effects of other mental states, and (3) behavioral outputs. In this context, two different *brain-state* tokens are said to be tokens of the same type of *mental state* iff they have the same causal relations to the input stimulus that the system receives, to its other inner mental states, and to its output behaviour. Any system, no matter what its physical realization, could have mental states provided only that it had the right causal relations between its inputs, its inner states, and its outputs. Finally, according to functionalist appraoch, if we would succeed in developing an artificial brain (computer) which would be functionally isomorphic to the natural brain, it/he would have also the same mental states. However, there are a few problems inherent to such an approach.

First, by raising the discussion to the level of *abstract functional structure*, functionalism neglects the fact that mental states are *primary qualities* of the specific *physical* structure brain; and there is no basis for believing that the abstract states which play the same functional role *in a different medium* would have the same primary qualities. Or, as Schweizer put it: "while there is good reason to believe that consciousness results from the complexity of the *physical* processes that take place in the brain, there seems to be no reason to conclude that different material implementations of the same *computational* structure will reproduce these same internal effects" [13, p. 272]. In other words, an approach concerned with the abstract functional structure of the human mind/brain could hope to replicate the abstract functional properties of the human mind/brain, but not the *real* one. Or: based on the idea of functional isomorphism, functionalism could be the right way to the smartest machines, but not to the conscious ("enjoying") one.

2.4 Language and Reality

Functionalists pass in silence over some basic conceptual problems, which makes their expositions less clear than it would be desirable. First, isomorphism is not identity, and isomorphic entities are not supposed to have the same properties but only the same structure: therefore, it is not sufficient to speak of functional isomorphism with human brain when aiming to replicate the human mental properties. Further, even the very idea of functional isomorphism is problematic, as long as we don't have a clear criterion on the basis of which we could decide when an artificial structure can be said to be functionally isomorphic with the human brain. Without such a criterion, all depends on the way one describes the brain: describe it in poor (reduced) terms, and you will easy construct an artificial system isomorphic to such a description! However, reality does not care much about our descriptions: hence, we could hardly replicate a phenomenon without knowing its real, and not merely "abstract", structure. Of course, the question of the real structure of the reality is not an easy one; in fact, it seems to be open-ended, and will probably remain such forever. It is immanent to science (in general) to search for such a conceptual system which would "carve nature at its systematic joins" [6, p. 279], but there is not much hope that such a result could ever be obtained. Consequently, every description of a phenomenon is inevitably dependent on the conceptual/categorial system of the beholder. However, although we cannot make a definite breakthrough to the Truth, we should at least know that expressions such as "all the control functions of the human brain" (from the above quotation), do not say much as long as we don't even approximately know what would count as

"all control functions". In other words, if "reproduction" (in that quotation) meant identical system, then the quoted assertion is trivial; if not, it is unclear.

3 Mind Is Less Than Machine

Despite the arguments stated above, there are ways to defend the position that mind is a machine; moreover, in the context of such positions, human mind is considered as a machine of rather humble abilities, which could be, and sooner or later will be, far outshone by artificial cognitive systems. Such views are based on some formal results concerning the computability, among which the most important are Universal Turing Machine and Church's Thesis.

The Universal Turing Machine is an abstract formal system which consists of a minimal number of symbols, states, and operations, in terms of which processes (algorithms) can be defined. The Universal Turing Machine can in principle be implemented on an ordinary digital computer. Church's Thesis says that every computable function is Turing computable. Roughly speaking, this means that any precisely defined process (algorithm) of symbol manipulation can be expressed by means of the Universal Turing Machine, and be carried out by a digital computer. Church's Thesis has not been proved because the concept of "precisely defined process" is not formally defined, but "it has been supported by evidence, much as any empirical scientific theory might be" [3. p. 66].

3.1 Simulation and Reality

Cognitive science should take into account the above results concerning computability. Namely, brain processes are natural phenomenon, and it seems reasonable to suppose that they can be described by a scientific theory; further, every scientific theory is essentially a syntactic system: consequently, brain processes should be describable in purely syntactic (computational) fashion. And according to Church's Thesis, that means that these processes could be carried out by a programmable machine, which should then have (or be in) also the same mental states as humans.

Opponents of such an interpretation of Church's Thesis will call our attention to the difference between simulation and reality; computer simulation of the process, they claim, cannot produce (create) real things/states. Simulations are based on some symbolic representations of reality, and all they can produce are new symbolic representations, but not real entities. For example, no computer simulation of the processes in the cow's udder could produce the real milk. The same, of course, holds for mental states: they can be simulated but not replicated by computational processes. However, there are reasons to hold (or hope) that with mental processes, things could be different. Namely, at least some human mental processes are algorithmic, and with it also literally reproducible by a computational machine. For example, there need not be any difference between the human process of carrying on an arithmetical operation, and the machine implementation of the same process: both, man and machine, could follow the same algorithm. Therefore, when mental processes are concerned, there need be no difference between the "reality" and the "simulation"; or, the human-brain computer and the digital computer can implement the same algorithm and obtain the same result.

3.2 The Language of Thought

It could be objected here that the given example seems too trivial to justify the great expectations immanent to the above line of thought. However, it seems reasonable to suppose (and to take as a working hypothesis) that there are many algorithmic processes which go on in human brains at the unconscious level, and which (if known) could be explicitly described in some formal language, and then also replicated by computer. And if it would turn out that all brain processes are algorithmic, the human brain - and *ipso facto*, human mind - would become completely machine (re)producible.

The first thing we need in the context of the above working hypothesis is some language in which we could describe the basic (unconscious) processes which take place in the human brain. Such a language could be Fodor's *language of thought*, which is taken to be common to all humans. An analogy with computers "is likely to be illuminating" here [9, p. 386]; namely, computers use (one or more) input/output languages by me-

ans of which they communicate with their environment and "a machine language in which they run their computations" [9, p. 385]: the language of thought is intended as a kind of machine language.

Fodor's proposal introduces an *intermediate level* between the physiological (hardware) and the conscious (input/output) level: cognitive processes are taken to be completely definable on this (intermediate) level. These processes are taken to be algorithmic, so that human thinking can be conceived of as computation over basic units (atoms) of the language of thought. It seems obvious that if such a brain/mind model could be the right one, than human cognitive abilities could be not only replicated, but far exceeded (in speed and scope/range) by programmable machines. However, some basic things concerning the above model are not yet clarified; first, Fodor's hypothesis imply the existence of a set of context-free atoms on which all our thoughts (i.e. computations) are based, and we cannot say what these basic atoms/units could look like. Clark describes such a (hypothetical) context-free atom as a "syntactic item" which "plays a fixed representational role", and as "an inner state which makes the same semantic contribution to each of the larger states in which it figures" [5, p. 31]. Such descriptions don't seems to be enough (in the operational sense); however, our actual ignorance concerning the "particulars" does not invalidate Fodor's hypothesis concerning the basic cognitive processes.

3.3 Language of the Mental

Fodor's proposal seems to be concerned primarily with thought processes; he stresses that "nothing can be expressed in natural language that can't be expressed in the language of thought" [9, p. 388]; but not all mental states (feelings, etc.) are really expressible in natural language. However, following Fodor's line of thought, it seems possible to make a move further and postulate the existence of a *language of the mental* (and a set of innate mental atoms as basic data items) by means of which all mental processes could be defined in the algorithmic fashion. In this context, we could paraphrase Fodor's assertion above by the following words: Nothing can be experienced by a human being that can't be expressed in the language of the mental. In other words, the hypothesis of the language of the mental would make the human mind completely definable at the syntactic level, and then, by Church's Thesis, also artificially reproducible by a programmable machine. (Results of syntactically defined processes are independent of the particular hardware.)

Of course, I cannot say how the language of the mental (and mental atoms) would look like; but we are in the same ignorant position also with Fodor's language of thought. However, in science we are often constrained to presuppose the existence of hidden and/or unobservable entities, structures, and processes, and then to judge the validity of such hypotheses by evaluating their formal consequences and empirical effects. Therefore, the fact that we cannot actually prove the existence of the language/atoms of some *basic mental level* should not be reason to abandon the very idea of the complete computational definability of the conscious human mind. Instead, if there is any real possibility that such a hypothesis could open the right way to the secret of the mental, we should proceed by it.

3.4 More Than A Natural Mind

A complete syntactic definition of the basic mental processes would have fascinating consequences. First, it would render possible the replication of any human mental state by running a computer program that implements a process which is *type-identical* to the process at the human brain's basic mental level which cause the given mental state in the human being. However, in such a case, it would become possible not only to replicate (imitate) human mental states, but also to produce *new kinds* of mental states, completely unknown to human beings. Namely, natural evolution (the selection principle) surely hasn't favoured the development of all possible (i. e., computable) kinds of mental states; on the other hand, an artificial brain/mind would freely explore a virtually open-ended combinatorial space, and so create completely new results (i.e., mental states). Finally, even only thanks to the enormously greater processing speed, such brain-machines would far exceed man's mental abilities. In other words, if we ever succeed to develop an artificial human-like conscious mental state based on pure computation, we should soon be faced with machines (artificial bra-

ins/minds) whose intellectual, emotive, and creative abilities will far excel those of the best among natural men. Some of the questions we should put to ourselves while moving in this direction could be: 'Who shall be then called to settle the measure of High and Low, of Good and Bad?'; 'Could such machines make a man better?'; and finally, 'Will not the natural mind, besides such artificial beings, become superfluous, or at least a servant to the proper product?'. (Un)fortunately, it seems that we have still time enough to think about such questions.

4 As You Like It

Independently of the position on the (possible) superiority of the programmable machine over the human mind, the computational model of the mind is widely used. However, mind and machines do not, in fact, have much in common. In this section I put forward three arguments for such a position, but I put forward also some reasons for the popularity of the computational model of the mind/brain.

4.1 A Commonsense Argument

The human mind/brain and programmable machines are completely different things on the physical level (brain/hardware) as well as on the psychic level (mental-states/software). Roughly speaking, the brain consists of a collection of neural networks which don't have anything in common either with serial (von Neumann's) or with parallel (Darwinian) type of hardware. Computers of the serial type consist of some basic units (processor, working memory, etc.), while in the human brain there is no similar physical component; on the other hand, parallel hardware consists of a bunch of serial computers which work in parallel mode, while brain's (sub)networks do not form anything similar to the bunch of (parallelly connected) serial computers. (More about it in the context of the third argument.)

Concerning the psychic/software level, the comparison is equally wrong. A program (serial and parallel) defines/creates a set of context-free data structures and a computational process (algorithm) over the (contents of) these structures, but there is no trace of mental states in all of this. On the other hand, it is well known that the human mind has very poor abilities of memorizing and performing algorithms except trivial ones. For example, many of us are not able to carry out a mental multiplication of two three-digit numbers.

It seems that these differences (complete discrepancy!) should be sufficient reasons for abandoning the computational model of the mind.

4.2 Being vs Assignment

One of the most discussed arguments against the computational model of the mind is Searle's *Chinese Room* thought experiment [14]. With this experiment Searle wanted to show that programs do not understand what they do/produce, independently of how intelligent their products/answers may seem to an observer. From that it should follow that programming is not the way which could lead us to machine understanding or mental states. I hold that such a conclusion is no less obvious without experiments than it is with them. Besides, Searle has interpreted wrongly his own experiment; namely, the experiment shows that the processor (Searle in the Room) does not understand, and not that the *program* (which Searle-processor execute) does not understand; but Searle will offer us a new (better) argument.

There are a few kinds of reply to the *Chinese Room* argument (catalogued in [14]); however, I hold that all these replies can be qualified as "arguments from ignorance". The most frequent among them (*System Reply*) emphasise the fact that there is not only a processor/program there, but a *whole system*: and it is possible that in some sophisticated systems some degree of understanding somehow simply emerges, although we actually don't know how and where. Of course, it is possible; however, it is equally possible that all actual computers are self-denying beings, which often suffer in silence, and sometimes make fun of us. But something of that kind doesn't seem plausible: hence, there is no much sense to put forward such kind of "arguments".

The *Chinese Room* argument argues that programs by themselves, as *purely syntactic* systems, cannot constitute mind. This argument "rests on the simple logical truth that syntax is not ... sufficient for semantics" [15, p. 200]. But in [15], Se-

arle put forward a new argument against the computational model of the mind, now based on the fact that "syntax is not intrinsic to the physics" [15, p. 208]. In short, the argument runs like this: (1) "computation is defined syntactically" (in terms of symbol manipulation); (2) "syntax is not intrinsic to physics" (an assignment of syntactic properties to physical phenomena is relative to an observer): consequently, (3) computational processes are *not intrinsic* to the physical world, and *ipso facto* not to the brain. In other words, it cannot be *discovered/shown* (as an empirical fact) that brain (or anything else) is intrinsically a digital computer; computational interpretation can be only *assigned* to the brain, as well as to anything else. And this then means that the assertion 'The brain is a computer' is not "simply false", but is "ill defined" and without a "clear sense" [15, p. 225]. Namely, if we interpret it as an assertion about the discovery of some intrinsic property of the brain, it is trivially false, while if we interpret it as a decision to assign the computational interpretation to the brain, it is trivially true/acceptable.

In sum, the results of Searle's two arguments seem to be the following: The *'syntax - semantics'* gap is fatal for the attempt to define the mind in terms of software, and the *'syntax - physics'* gap is fatal for the attempt to qualify the brain as computational hardware. In other words, we don't have stronger formal reasons to conceive of the human mind/brain as a kind of computer than we have it for anything else: therefore, the computational model of the mind/brain says us, in fact, nothing essential/intrinsic of the mind/brain.

4.3 Custom and Necessity

The standard computational model of the mind is essentially of Fodor's type: it presupposes the existence of a stock of context- free syntactic atoms (as physical tokens and content bearers) upon which the cognitive (algorithmic) operations are performed. Atoms form the combinatorial base of all potential thoughts, and occur unchanged across all the cognitive processes. Human beings are supposed to inherit a fixed set of such atoms, while the learning and inference processes consist in the "recombination and redeployment" of the preexisting context-free representational primitives [5, p. 225].

A completely different cognitive model has been developed by the *connectionist approach*, which is based on results of neurophysiology. Roughly speaking, the human brain is a set of neural networks; such networks learn (acquire knowledge) by repeated exposure to a training environment (which includes some form of feed-back effects); a network starts with an innate/random distribution of its hidden unit weights, while exposure to a given environment causes it to permanently adjust its connection weights in a way which tend toward the best output.

The network hardware differs not only from the hardware of classical (serial) computational systems, but also from that of parallel (PDP) systems; namely, although it is common to describe the human brain as "a massively parallel processor" [6, p. 156], there are, in fact, no "processors" in the neural networks neither in natural, nor in artificial ones. Further, there are no "programs" in connectionist systems: such systems are trained and not programmed. Finally, there are no context-independent "data atoms/records" in connectionist systems, because these systems are based on *superpositional representations* of knowledge/data. The representation of knowledge K1 and K2 is said to be superposed if K1 and K2 are represented by the same resources. In superpositional systems there are no context-independent data records which could be said to stand for ordinary semantic units (data and propositions) because every piece of knowledge is stored holistically in the sense that it is distributed throughout the network: any weight (in the network) can take part in the encoding of any piece of knowledge contained in the network. And if a given (trained) network is later trained to learn/accept a new piece of knowledge, the existing weights (although preserving previous knowledge) will be changed: hence the context-dependence of the data records (i.e. of knowledge representation) in superpositional systems.

Flanagan holds that the connectionist models "have called the distinction between software and hardware into question" [8, p. 180], while Clark says that connectionist models are characterized by "the lack of a firm data/process separation" [5, p. 14] because such systems do not involve program-driven computation over a fixed set of

symbols. Indeed, there are in fact neither "symbols" nor "programs" nor "processors" in connectionist systems. But why then at all use the standard computational taxonomy here? And although virtually all authors emphasize the essential difference between standard computational models and connectionist models, they keep on using standard computational taxonomy (see, for example, [5, 6, 8]). It seems that there are two main reasons for such praxis.

The first reason could be *custom*; namely, in spite of all the differences, connectionist systems still have some inputs and outputs, and some processes in between; therefore, one is inclined to use the same old terminology. In this context, rather than change the terminology, Clark criticizes the "tendency to identify foundational computational ideas too closely with their particular incarnations in classical systems". He holds that "it may be more productive to seek less restricted understandings of such concepts - understandings which can cut across many types of computational device (connectionist, classicist, and types as yet undreamed of)" [5, p. 122].

Therefore, connectionists simply want to preserve the computational terminology; however, they are in a way also constrained to do so. Namely, with the superpositional knowledge representation, a weight (or set of weights) cannot be identified with any fixed semantical concept/content because each weight in the network contributes to the representation of many such semantical units. Analogous problems characterise attempts to explain processes which take part in the network, its skills/abilities and the results it produces. In short, connectionists need something by means of which they could form/express a kind of *top-level explanation* of what is going on in a network, because without such an explanation their results and working methodology would be "obscure" [5, p. 49], or at least not of the scientific kind. And the standard computational terminology seems to be an appropriate means for all such explanation. There are various techniques/methods by which such top-level explanatory schemes are developed (see, for example, [5]); however, we must know that such top-level descriptions are only post hoc semantical explications of what the network knows/does, and not of what is really going on inside the connectionist system.

To sum up, with all types of cognitive models, we use/need more levels of functional descriptions of the phenomena on which we try to identify (model and handle) those features of the mental which we find interesting. But we should not confuse reality with descriptive models; it can be useful/necessary to assign the standard computational model to the natural systems of neural networks as well as to artificial ones, but there is all the difference between the "deep reality" of such systems and their top-level algorithmic descriptions.

5 When You Need It

I agree with Searle that mind is not intrinsically a digital computer, but I hold that his criticism partly misses the point. Namely, virtually nothing can be said to be "intrinsic to the physics", for virtually everything that "exists" (entities, forms/kinds, qualities. etc.) has been assigned to the physics by the observer/interpreter: therefore, it would be pretty hard to hold that brain literally is a digital computer. However, Searle leaves opened the main question we are concerned with here, and that is: Can the (unknown) relation between the mental/mind and the physical/brain be successfully studied/explained on the basis/model of the (known) relation between software and hardware? And in the context of this question, the fact that "nothing is intrinsically a digital computer" [15, p. 212] counts rather little. Finally, we should put also the question why is the computational model of the mind so widely used if it is not valid. But the answer to this question seems rather simple: we need some cognitive model, and we don't have a better one. Namely, "no one has much of a clue" [5, p. 224] about the nature and the ways of emerging of conscious mental states: consequently, we are constrained to speak "in figures", and the most appealing set of figures seems to be the one offered by the computational technology.

5.1 The First Move

For Clark it is a "mystery ... how conscious content is possible at all" [5, p. 224], while Dennett qualified human consciousness as "just about the

last surviving mystery"; he defines 'mystery' as "a phenomenon that people don't know how to think about" [7, p. 21]. And in such cases, figurative speech enters the stage as the only possible tool of thought and of creative imagination. If a theory in general can be said to be "the conceptual vehicle with which we ... come to grips with the world" [6, p. 117], we could say that metaphor is a vehicle with which we come to grips with the inexpressible: it is the first move toward a scientific theory.

Metaphor is a mapping between the two domains, and as a such it forms a cognitive model which give us an opportunity to speak of one (unknown) domain in terms of another (known) domain. Black [2] speaks of the isomorphism between two domains of the metaphor, while Lakoff stresses that the mapping defined by metaphor must "preserve the cognitive topology of the source domain, in a way consistent with the inherent structure of the target domain" [11, p. 215]. Let us note that isomorphism by itself does not guarantee the preservation of the cognitive topology because an isomorphic mapping could be defined in a way which is not in accordance with the cognitive topology. Further, most metaphorical mappings are not isomorphic, but partial and/or incomplete in the sense that not every entity from the source domain has its counterpart in the target domain, and not every entity from the target domain has a counterpart in the source domain. In this context, the degree of the preservation of cognitive topology and of the isomorphism of structures can be taken as the criteria of the strength/validity of the metaphor: the strength of metaphor rests on its "systematic structural match between the two domains" [10, p. 453].

According to Boyd, the computational metaphor of the mind has an "indispensable role" in the formulation of theoretical positions in cognitive science, and has provided "much of the basic theoretical vocabulary of contemporary psychology" [4, p. 487]. The impact of figurative speech on cognitive science seems to be really immense; indeed, the greatest part of the ideas are expressed in figurative fashion, and disputations are often (only) wars with metaphors. As an indicative example, let us mention the concluding paragraph of Dennett's extensive book; he admitted that his explanation of consciousness was "far from complete"; namely, he has not proposed a new scientific theory, but only a new metaphor. "All I have done", he says, "is to replace one family of metaphors and images with another" [7, p. 455]. Therefore, faced with the mystery of conscious mental states we are, in fact, still on the first move. But there have been made attempts to go further, fast and far.

5.2 A Going Beyond

The computational metaphor of the mind emphasises various (alleged) features of the *'software - hardware'* relation, some of which took the form of the basic working principles in cognitive science. One of the most influential of these features is the *independence* of the software from the hardware. By analogy, it is taken that mind should be independent of the brain, and consequently that: (1) mind can be studied independently of the brain, and (2) mind can be realized by means different than the human brain. I hold that both the above hypotheses are worthy of research; however, the independence of software from hardware should be well understood before putting too great expectations on it; let us see an example. Dennett says: "if what you are is the program that runs on your brain's computer, then you could in principle survive the death of your body as intact as a program can survive the destruction of the computer on which it was created and first run" [7, p. 431]. As is often the case with Dennett's arguments, I must say "perhaps"; namely, to see the real strength/weakness of the present argument we should first clarify (1) under which conditions can a program "survive the destruction of the computer", and (2) how could we afford the same conditions for the human mind (understood as a program).

In its *source form*, a program does not depend on hardware, but it depend on the compiler for (on top of) which it was written: the destruction of the compiler would make a source (not compiled) program "dead". Namely, a source program for which there is no (more) compiler is but a heap of signs without meaning, because it is the compiler that defines ("gives life" to) the syntax and semantics of a programming language, and with it, to all programs written in that language. On the other hand, after being *compiled* (linked and loaded), the program no more needs (depends on)

the compiler, but is now dependent on the given hardware (on/in which it has been loaded). Moreover, when a program is compiled, loaded and linked, it could be conceived of as "a part" of hardware; namely, what in the source program were words and sentences (instructions) are now simply energetic (tensional) states of some points (bits) of the hardware. But couldn't a program, even in such a form, be copied on a new hardware (and so outlive the old one)? Perhaps; but without additional adjustments, only on an "nearly identical" one. Therefore, for the survival of the human mind (after the death of his brain-hardware) we should have a "nearly identical" new brain, and "a version of functionalism" (Dennett's position) will not afford us anything of that kind.

To sum up, it is of little avail to try to map the hardware independence from the domain of computer system on the domain of human mind/brain system as long as we don't know (notably) more about the lower levels of the latter. Hardware independence has been developed with an essentially *bottom-up* approach: one must know the below level to develop an interface which makes the higher level entities (relatively) independent from those of the lower level. On the other hand, to deal with the mind on the abstract/functional level, while leaving aside the physical idiosyncrasies of the brain, means to follow the *top-down* approach. And it is hard to expect that such an approach could lead to some spectacular results concerning independence before we touch the bottom/brain. In other words, the computational metaphor/model of the mind is of a limited power; it could be useful inside some limits, but when we try to step by it further than it can lead us, we bind our efforts to failure.

6 Conclusion

Theories/paradigms are constructions rather than discoveries; the phenomena can be described in endless ways; the Truth (if it exists) is unattainable, so that the *pragmatic value* (if we can *recognize* it) remains our only guide in the scientific enterprise. Processes in the human brain are not intrinsically computational; however, it is a common scientific practice to try to apply a known models to a new (unknown) domains; such is also the attempt to explain the human mind/brain on the basis of the computational model. I hold that such an approach can give good results of the *functional* type, but concerning the phenomena of *consciousness* we are still in the scope of speculations. And if we are ever to reveal the mystery, the connectionists approach could be the best one. However, it seems that there can be no real progress on the way to artificial mental states as long as we don't know more about the ways the mental states emerge in the human brain.

References

[1] B. Beakley & P. Ludlow (eds): *The Philosophy of Mind*, The MIT Press, 1992.

[2] M. Black: 'More about metaphor', in A. Ortony (ed).

[3] S.G. Boolos & C.R. Jeffrey: *Computability and Logic*, Cambridge University Press, 1980.

[4] R. Boyd: 'Metaphor and theory change: What is "metaphor" a metaphor for?', in A. Ortony (ed).

[5] A. Clark: *Associative Engines: Connectionism, Concepts, and Representational Change*, The MIT Press, 1993.

[6] M. P. Churchland: *A Neurocomputational Perspective: The Nature of Mind and the Structure of Science*, The MIT Press, 1992.

[7] D. C. Dennett: *Consciousness Explained*, Penguin Books, 1993.

[8] O. Flanagan: *The Science of the Mind*, The MIT Press, 1992.

[9] J. A. Fodor & Z. W. Pylyshyn: 'How There Could Be a Private Language and What It Must Be Like', in B. Beakley & P. Ludlow (eds).

[10] D. Gentner & M. Jeziroski: 'The shift from metaphor to analogy in Western science', in A. Ortony (ed).

[11] G. Lakoff: 'The contemporary theory of metaphor', in Ortony, A. (ed).

[12] A. Ortony (ed): *Metaphor and Thought*, Cambridge University Press, 1993.

[13] P. Schweizer: 'Intentionality, Qualia, and Mind/Brain Identity', in *Mind and Machines*, 1994, Vol. 4, No. 3, pp. 259-282.

[14] J. Searle: 'Minds, Brains, and Programs', in D. Hofstadter & D. C. Dennett (eds): *The Mind's I*, Penguin Books, 1981.

[15] J. Searle: *The Rediscovery of the Mind*, The MIT Press, 1992.

What Internal Languages Can't Do

Peter Hipwell
Centre for Cognitive Science
University of Edinburgh
E-mail: petehip@cogsci.ed.ac.uk

Keywords: language, analogy, emergence

Edited by: Matjaž Gams
Received: May 15, 1995 **Revised:** October 25, 1995 **Accepted:** November 29, 1995

The ability of artificial internal languages to mirror the world is compared to the power of natural language systems. It is concluded that internal languages are equally as arbitrary, and therefore have no representational advantage. Alternative forms of representation, including particle interaction in cellular automata, are considered.

1 Introduction

There is a prevalent AI assumption that natural language is, in some way, messy and entangled, and that it must be mappable to some other language form – an internal, regular, logical form which is somehow reflecting the actual structure of the world in a more straightforward manner.

This paper suggests that such internal language systems cannot exist in the way conceptualized and that any language-like description of mental states (especially where pertaining to "word meaning") may well be misdirected. The assumption that natural languages are fuzzy and biased descriptions of the world may be correct, but this does not mean that there are better descriptions in terms of better languages.

The problem arises from an intuitive picture of the relation between computer languages, Turing Machines and human thought. The metaphor is very possibly a misleading one. We assume that neurons can be modelled, because they are carrying out deterministic, computable processes. But there is no guarantee that there is a parsimonious higher level redescription of brain states (i.e. groups of neurons, or other simple computing elements) in terms of a language other than a natural language.

2 Saussure's Approach to Language

De Saussure (1983) claimed that "Everything having to do with languages as systems needs to be approached ... with a view to examining the limitations of arbitrariness"

In considering the relation between the surface form of natural language and the mental representation of its meaning, de Saussure noted that the mapping from the surface form of individual words to their meaning is for the most part arbitrary. That is, there is no substructure within the surface form (signifier) that allows its relationship to the underlying meaning (signified) to be deduced. There are minor exceptions to this general principle, such as inflectional structure and onomatopoeia.

These examples do not serve to undermine the general conclusion: word meanings have to be learnt rather than inferred. This position is an 'anomalist' view of meaning. The alternative view, that a language may have basic elements which are similar in structure to whatever is being represented we will call an 'analogist' position.

De Saussure differentiated between signifiers being totally arbitrary, and the relative arbitrariness of constructions such as 'dix-neuf' – nineteen in French, composed of 'dix' (ten) and 'neuf' (nine). Although the meaning is derived from the components, the ordering of the elements and the way in which meaning of the whole is derived is

also arbitrary. However, once the pattern for deriving the meaning of the compound has been learnt, then all such patterns can be interpreted.

There is no absolute division between what is arbitrary and what is analogous: take for example a visual scene, a photograph of the scene, a sketch of the scene, a linguistic description of the scene, and a single word. Analogies can be drawn between each of these; different levels of detail and structure are available in each case. Although the linguistic example does not explicitly look the same as the visual scene, it is important to bear in mind that the mind must be able to construct the linguistic structure from the visual scene - therefore in some sense it is an analogy. But *any* mapping process could be said to create similar or analogous structures: when we cannot trace the process, we can call the relationship "arbitrary".

3 The Medium Of The Mind

A common AI description of the "medium of the mind" is one involving language-like representations. That is, we have arbitrary symbols (word or morpheme analogues) which are combined by some mechanism (thought) to give meaningful expressions (which can often be directly related to states of the world). This is the basis of Newell and Simon's definition of physical-symbol systems (Newell and Simon 1976).

In the Saussureian picture thought is too formless for study – there is no object of study outside a language itself. However, when it is held that regular internal languages are superior to natural language descriptions precisely because the structure can clearly be related to the world – so an analogy to objects and their relations would be expressible in a perfect language of thought – we have to ask exactly *what* is being contributed by the artificial system. Such languages are relatively arbitrary just like a surface language form.

The power of the artificial systems is based on combinatory and modification mechanisms which should be constructed so as to allow for correct inferencing and therefore intelligent reasoning; a mirror of the world. As Fodor and Pylyshyn (1988) emphasize, it is systematicity and productivity that underlie the power of such language systems. But these are properties of natural language as well. Although it is assumed that our grammar systems are generally not *punctuate*, certain complex properties, such as the existence of non-literal language and fuzziness of descriptions make it messy to formally describe.

Coherent models using internal languages range from GPS (Feigenbaum and Feldman 1963) to SHRDLU (Winograd 1972, 1980) to the naive physics project (Hayes 1979, McDermott 1987) and beyond. However, many of these research programs have run into difficulty. I would like to suggest that this is because there is something wrong with this picture; namely that there is no fundamental difference between a 'messy' natural language and a 'clean' artificial one, in terms of the analogy properties that are required.

4 Problems with Analogy Languages

The idea of an analogical language composed of arbitrary units is problematical because natural language then is equally powerful. This problem is neatly summed up by a quote from Wittgenstein (1974): "The rules of grammar cannot be justified by shewing that their application makes a representation agree with reality. For this justification would itself have to describe what is represented. And if something can be said in the justification and is permitted by the grammar - why shouldn't it be permitted by the grammar I am trying to justify?"

In other words, no language can be used to justify another in terms of giving a better description of reality, simply because no language can be said to have a better syntactic mechanism for doing this.

We might argue that the internal representation does not have to be in a language-like form. De Saussure gives the example of a picture dictionary as the common model people have for the way in which word meanings are stored. So, for Latin, the surface form for tree is *arbor*, the mental representation is a picture of a tree, for a horse *equus* and the relevant picture. So we seem to have some kind of analogistic representation.

But there is absolutely no justification for the picture-dictionary model to be taken as a more appropriate version of the mental representation of meaning. The fact that we introspect visual images is of no import. We are still left with

the problem of working out how these entities are combined in thought; imagine trying to combine the pictures from the picture-dictionary together to give a picture of a sentence meaning. The only medium we know of where this type of combination is natural and obvious is in common-sense. And common sense is notoriously difficult to formalize.

The problem is especially clear when we look at natural language, and see that we already have one level of anomalous representation (surface language) and one level of analogous representation (sensory transduction). We can imagine a chain or gradient of arbitariness between them - but the artificial language approach doesn't seem to fit anywhere in the chain other than at a point equivalent to natural language.

A similar regress argument is touched on by Dennett (1991) who points out that a mentalese (internal language) will not help in any way in explaining how we conceptualize things we are going to say – because then we have to explain how conceptualization of the mentalese sentence occurs, and so on.

This leaves the internal language advocate with a number of problems:

- Where do arbitrary primitives come from? If they are arbitrary how can they represent things other than by a direct link to those things? (This is really a variant of the symbol grounding problem (Harnad 1990) and also applies to natural language).
- Where does the extra power of an internal language come from?
- Why is it that the primitives and combination of an internal language are correct, whereas natural language words are not?
- Why don't we communicate in more perfect forms of language - a whim of evolution?

Another possibility is that there is a simpler analogy representation that surface language can connect to: perhaps something along the lines of the $2\frac{1}{2}$D and 3D sketches (Marr 1982). This seems more tenable, because analogy is not the same as isomorphism, so we have the option of dropping certain pieces of irrelevant information or noise. We also have plenty of information about the way the brain processes data that support this (for example, multiple topographic maps at different depths of analysis). However, there is the danger of falling into a picture-dictionary analysis of meaning. The chain of analogies must be explicitly traceable the whole way to language.

The final possible argument is that sensory transduction is not analogous to anything: if we regard sensory data as being like words, then perhaps we can regard all representations as being relatively arbitrary all the way up.

In fact, the sensory array simply *can't* an arbitrary code. An iconic (analogous) sign is somehow structured like the thing it is conveying. Naturally, we don't see much similarity between a list of neural firing rates and the structure of the world that we perceive. But making a comparison of this nature is erroneous, because sensory transduction is (for this purpose) the place where all explanation must come to an end. This is all that we experience - it *is* the raw structure of the world.

5 Non-Language Systems

Von Neumann (1958) wrote that "[L]ogics and mathematics in the central nervous system, when viewed as languages, must structurally be essentially different from those languages to which our common experience refers." Systems which do not use the traditional forms of formal language are beginning to be utilized.

5.1 Physical Symboloids

Van Gelder and Port (1993) discuss the possibility of a range of physical "symboloid" systems. These are defined by three points:

1. There is a set of primitive types Pi; for each type, there is available a potentially unbounded number of actual physical instances or tokens (symboloids);
2. There is a (possibly unbounded) set of compound types, Ri; likewise, for each type, there is available a potentially unbounded number of actual physical instances or tokens; and
3. There is a set of transitive and non-reflexive constituency relations over these primitive and compound types.

Traditional models concentrate on static concatenative combination of static arbitrary primitives. Static concatenation is the combinatory process of the written form of language, where symbols are arranged in a linear sequence, maintaining their identity within the composed representation. This approach conceals a number of assumptions about the possible form of the components and the ways they can be combined. Other systems may contravene these assumptions:

Primitives

- May be either static like written words or dynamic like spoken words.
- They could be analog or digital - there may be a continuum of possible token types.
- Configuration may not be arbitrary.

Combinations

- The process of combination can have dynamic or static aspects. For example, dynamic primitives can be combined statically – one example is a musical chord. Static primitives can be combined dynamically – the primary structure of a protein is determined by the sequence of amino acids, but as the chain forms, other interactions such as the formation of hydrogen and van der Waals bonds between distant acids causes the protein to take a secondary and tertiary structure.
- The mode may have one of three effects on the primitives (these are not sharply differentiated). The primitives could simply be concatenated, or they could be combined in a context-sensitive manner, giving distorted but recognizable symbols, or they could be combined in a functional manner, which doesn't preserve constituent structure.
- The syntactic rules for composition can be more or less strictly applied (ie there might be a certain degree of fuzziness in rule applicability or ways in which the rule is applied).

Van Gelder and Port conclude that it is not at all clear which of the possibilities are actually employed in cognition. For the most part it is hard to work out where they should be applied, and what advantages they might give. However, it is certainly true that these possibilities have, for the most part, been neglected in AI research, because they are not so easily programmed in traditional computing architectures.

5.2 A Concrete Example Of Emergent Symboloids

Recent computer science paradigms such as evolutionary computing (Koza 1992) and computation by cellular automata (Michell, Crutchfield and Hraber 1994) give a powerful way of coming up with representations and processes corresponding to the symboloid description.

In these approaches, the primitives and methods of combination are fixed by the programmer (e.g. LISP operations and syntax, mapping templates), and although structures can evolve at a higher level, these basic level elements must remain. There is no way around this because there will always have to be a metarepresentational format to represent the representations in (in the case of brains, this is the behaviour of neurons). However, the behaviour of the overall system may not easily be describeable in terms of the primitives and combinatory rules. The notion of *emergence* of system properties does not rely on a description in terms of arbitrary symbols. How and when emergence occurs is an interesting current research problem: notions such as whether computation in cellular automata occurs at "the edge of chaos" (Langton 1990, Mitchell, Hraber and Crutchfield 1993) are being disputed.

Cellular automata (CA) are perhaps the simplest kind of system to look at. These are composed of cells (often a one dimensional row of cells), which change state over time. This is decided using a rule which determines what state the cell switches in to, given any pattern across a local set of cells (the template). The update rule is usually the same for every cell. To analyze the emergence of globally organized behaviour in CA systems which have been evolved to perform specific tasks, it is possible to filter out "particles" and trace the interactions between them. This kind of work is described in Hanson and Crutchfield (1992).

The system described in the paper is used to classify an initial configuration in terms of proportion of cells turned to zero at the start. If this

is more than half the cells, then the final output (after a fixed number of iterations) should be all zeros. If there are less than half at zero, the output should be all ones.

The particles are discontinuities between computationally homogenous regions of the array that are found in some runs that settle on sophisticated processing strategies. Different particles may meet each other as they move across the array over time: they interact in a variety of "symboloid" ways - they may cross each other withough interference or may interact to annihilate each other, they may react to produce new types of particle, or they may simply decay. These non-compositional and non-reversible interactions carry out a computational function, allowing information to be transmitted between different regions of the array, leading the system to construct its final classification.

This is probably the best explicit example we have of interacting symboloids which underlie a symbolic (arbitrary) classificatory system (even though the example in the paper is only performing a binary categorization). We start the CA with an input pattern (which we might think of as being comparable to a sensory array), for which it has to produce an arbitrary output. The symboloid structures are not analyzable in terms of their similarity to the full scale patterns,

Given the arguments against language-like internal representations, it seems likely that these new methods for computation, although relatively novel and primitive at present, will have a significant role to play in future AI models.

Furthermore, the CA can be described in terms of a Mealy Machine (formal language rewriting system). This fits in with the assumption that, at some level, all processes can be described in terms of formal language systems.

5.3 Connectionism

The more familiar modelling paradigm of connectionism can also support these arguments. For example, the Recursive Auto-Associative Memory (Pollack 1990) also instantiates alternative forms of compositionality, being able to represent tree structures on a fixed number of units. The compositional structure does not preserve the characteristics of its components.

Clark (1993) argues that in connectionist systems there doesn't have to be any lowest level of context-free atoms (arbitrary primitives); in some cases you can't actually analyse out that there are primitives. We can't rely on the activations or weights of a connectionist net or the particles of a CA process as context-free units – these are not necessarily interpretable on their own, and do not have to be the same in variants of the same system trained/evolved from different starting points.

This is also true of real brains. Even sensory transduction is not context free - although the responses of individual transducers are deterministic, they are also highly ambiguous in terms of relation to external stimuli. This ambiguity is present at various levels in the visual cortex (de Yoe and van Essen 1988), the most closely studied cortical processing area. The presence of ambiguous response (or coarse coding) can allow for a better overall encoding than having a greater number of more highly tuned transducers (Hinton, McClelland and Rumelhart 1986). This principle means that we don't have to have a lowest level at which context-free arbitrary primitives are employed.

6 Conclusion

This line of thought is particularly salient to one classical attack on the hard AI paradigm: the "Chinese Room" argument (Searle 1980). The thrust of this argument is that arbitrary symbols are not sufficient to encode meanings. This clashes with the idea that Turing Machines, which use wholly arbitrary symbol manipulation, are able to perform any computational process (including those of the mind).

The confusion arises between different levels of description: a description in terms of language-like representations can be made. But from the arguments discussed, it seems that internal languages do not underlie natural language characteristics. Emergence of properties, such as the arbitrary referential nature of words, do not have to rely on the presence of similar underlying structures – indeed, to claim such is an argument similar to the homuncular theory of visual perception.

References

[1] Andy Clark (1993) *Associative Engines*, MIT

Press.

[2] F. de Saussure (1983 translation) *Course in General Linguistics*, London: Gerald Duckworth.

[3] E. A. de Yoe and C. D. van Essen (1988) Concurrent Processing Streams in Monkey Visual Cortex, *Trends in Neuroscience*, 11.

[4] Daniel Dennett (1991) *Consciousness Explained*, London: Penguin Books.

[5] Edward A. Feigenbaum and Julian Feldman (1963) GPS, A Program that Simulates Human Thought, *Computers and Thought*, New York: McGraw-Hill.

[6] Jerry Fodor and Zenon Pylyshyn, (1988) Connectionism and Cognitive Architecture: A Critical Analysis, *Cognition*, 28.

[7] Patrick J. Hayes (1979) The Naive Physics Manifesto, *Expert Systems in the Micro-Electronic Age*, ed. D. Michie, Edinburgh University Press.

[8] James E. Hanson and James P. Crutchfield (1992) The Attractor-Basin Portrait of a Cellular Automaton, J. Stat. Phys. 66:1415.

[9] S. Harnad 1990 The Symbol Grounding Problem, *Physica D*, 42.

[10] G. E. Hinton, J. L. McClelland and D.E. Rumelhart (1986) Distributed Representations, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition vol 1*, ed.D. E. Rumelhart, J. L. McClelland and the PDP Research Group, MIT Press.

[11] John R. Koza (1992) *Genetic programming : on the programming of computers by means of natural selection*, MIT Press.

[12] Chris G. Langton (1990) Computation at the edge of chaos: phase transitions and emergent computation, *Physica D*, 42.

[13] Drew McDermott (1987) A Critique Of Pure Reason, *Computational Intelligence*, 3.

[14] David Marr (1982) *Vision*, San Francisco: W. H. Freeman.

[15] Melanie Mitchell, Peter T. Hraber and James P. Crutchfield (1993) Revisiting the Edge of Chaos: Evolving Cellular Automata to Perform Computations, *Complex Systems*, 7.

[16] Melanie Mitchell, James P. Crutchfield and Peter T. Hraber (1994) Evolving Cellular Automata to Perform Computations: Mechanisms and Impediments, *Physica D*, 75.

[17] Allen Newell and Herbet A. Simon (1976) Computer Science as Empirical Enquiry: Symbols and Search, *Communications of the Association for Computing Machinery*, 19.

[18] J. Pollack (1990) Recursive Distributed Representations, *Artificial Intelligence*, 46.

[19] John R. Searle (1980) Minds, Brains and Programs, *Behavioural and Brain Sciences*, 3.

[20] Timothy van Gelder and Robert Port (1993), Beyond Symbolic: Prolegomena to a Kama-Sutra of Compositionality in *Symbol Processing and Connectionist Models in Artificial Intelligence: Steps Toward Integration*, ed. Vasant Honavar and Leonard Uhr, Academic Press.

[21] John von Neumann (1958) *The Computer and the Brain*, New Haven: Yale University Press.

[22] Terry Winograd (1972) *Understanding Natural Language*, New York: Academic Press.

[23] Terry Winograd (1980) What Does It Mean To Understand Language?, *Cognitive Science*, 4.

[24] Ludwig Wittgenstein (1974) *Philosophical Grammar*, Oxford: Blackwell.

Consciousness and Understanding in the Chinese Room

Simone Gozzano
via della Balduina 73,
00136 Rome, Italy
E-mail: s.gozzan@phil.uniroma3.it

Keywords: Searle's Chinese room

Edited by: Matjaž Gams
Received: April 26, 1995 **Revised:** October 27, 1995 **Accepted:** November 10, 1995

In this paper I submit that the "Chinese room" argument rests on the assumption that understanding a sentence necessarily implies being conscious of its content. However, this assumption can be challenged by showing that two notions of consciousness come into play, one to be found in AI, the other in Searle's argument, and that the former is an essential condition for the notion used by Searle. If Searle discards the first, he not only has trouble explaining how we can learn a language but finds the validity of his own argument in jeopardy.

In the well-known "Chinese room argument," John Searle argues against the idea that the process of understanding a language can be tantamount to mere manipulation of formal symbols. Over the years the argument, considered fatal against "strong Artificial Intelligence," has provoked a number of objections (see commentary to Searle 1980; Carleton 1984; Rey 1986). Here I shall present another possible one. My hope is that this objection will shed some light on the relationship between understanding and consciousness. As I will argue, Searle's position assumes that to understand a sentence one must necessarily be conscious of its content in a way I will specify later. So, the purpose of this paper, then, is, first, to differentiate understanding from being conscious of understanding and, second, to clarify the role that various notions of being conscious play in the argument. In this way I hope to show that in both cases Searle fails to make his point. Let's start with the argument itself. Searle's original purpose was to demonstrate that computer programs, however complex and accurate, will never be able to understand a language. Simplifying somewhat, Searle's argument goes as follows: Searle, completely ignorant of Chinese, is locked into a room with a book of Chinese symbols and a book, in English, that explains how to combine and transform the Chinese symbols. Every now and then, sheets of paper with Chinese symbols written on them are slipped to him from under the door. His task is to give back a sheet of paper with Chinese symbols whenever he receives one. To do this, he compares the symbols on the incoming sheet with those on the Chinese symbols book, checks which rules are allowed for the occurring symbols, and transforms them accordingly. In this way, Searle transforms the symbols by means of a set of rules in a purely "formal" way, that is, by identifying the symbols just by looking at their shapes. Outside the room there is a Chinese person who is giving the sheets to Searle taking them to be "questions" and the sheets handed over by Searle to be "answers." As a matter of fact, the Chinese person, a perfectly fluent native speaker, considers Searle's answers to be adequate responses to the questions. Consequently he believes that inside the room there is somebody who understands Chinese, and grants in this way that Searle has passed the Turing test for Chinese. However, Searle's comprehension of Chinese is not improved by his symbols processing. Hence, understanding a language is not equivalent to symbols processing, and the Turing test is not sufficient to determine understanding (Searle 1980). Against this argument a number of objections are possible. The most interesting one is the so- called "systems reply." According to

this objection, even if Searle himself does not understand Chinese, the entire system, that is, the books, the room, the execution of the functions, etc., does. Searle has a rebuttal to this line of thinking: even if he memorizes all the contents of the books and the rules, and walks around uttering Chinese, he still wouldn't be a Chinese speaker, insofar as he would not understand Chinese. Let's inspect the main argument and this rebuttal more closely. Consider the main argument from Searle's point of view. It goes as follows (1):

i) I'm manipulating formal symbols for Chinese ii) I do not understand Chinese iii) The manipulation of formal symbols is not equivalent to understanding

This very argument has been transposed, by Searle himself, also in terms of syntax versus semantics (Searle 1990). The idea is that we may substitute "manipulation of symbols" with "syntax" and "understanding" with "semantics". Here is the new version of the argument:

i) I'm doing syntax for Chinese ii) I'm not semantically competent with Chinese iii) Syntax is not sufficient for semantic competence

Searle's conclusion, in this new formulation, is that syntax is not sufficient for "taking care" of semantics (Haugeland 1981). Now, how does all this demonstrate the insufficiency of the Turing test as a test for linguistic competence and, hence, for understanding? We saw that, according to Searle, purely syntactical manipulation is sufficient for passing the test. So, in order to pass the Turing test it is not necessary to have semantics, that is, to have an intentional mind. Now, consider the problem of being a judge for a Turing test. The judge is a normal human being that, at the end of the test, supposes that there must be a competent Chinese speaker inside the room or, in the systems reply, that Searle is a competent Chinese speaker. On what basis could the judge evaluate the adequacy of the responses by Searle? Given that he evaluates not only the syntactical correctness of the responses, but also – and primarily – their semantical adequacy to the questions, his judgments must be grounded in a semantical basis too. But since what Searle is doing is nothing but symbol manipulation, we conclude that syntax is sufficient for semantics. Therefore, in order to show that syntax is not sufficient for semantics, Searle has to suppose that syntax is sufficient for semantics. How is such a simple rebuttal of Searle's argument possible? It seems to me that *two different notions of semantics* are at stake. ¿On one side, say from the Turing test perspective, Searle may manage reference - for instance, he may perform correctly on questions like "could you indicate a red jacket?" - and truth - correctly replying to a list of true/false questions. On the other side, say from the first person perspective of Searle himself, he does not know what he is doing; he cannot, as it were, "inspect" the contents of its own utterances. Analogously, since semantics was intended as a substitute for understanding, we have two notions of understanding either. ¿In one case, that of the Turing test, Searle does understand; in the other, that of the first person perspective, Searle does not understand what he is saying or doing. This second notion, however, is not the notion of understanding *per se* , but the notion of being conscious of understanding. Searle's argument rests on the idea that understanding a language necessarily implies being conscious of the contents of utterances or mental states. It is Searle's task to show that understanding necessarily implies consciousness. Recently Searle has argued exactly along these lines, claiming that we must be able to be conscious of all the mental contents we have, at least in principle (Searle 1992). Now, what relationship between understanding and consciousness may we have in AI? Consider the case of AI program SHRDLU. SHRDLU simulates a robot arm which can move a number of solids, such as cubes, pyramids and spheres, in a fictional world, that is, in a world completely generated by the computer itself (Winograd 1972). A human being gives SHRDLU commands such as "pick up a pyramid and put it over a big blue block," and SHRDLU reports what it is doing and why. Since SHRDLU may report what its final task is, and what the relevant steps it has to perform to accomplish the task are, I submit that it is conscious of what it is doing in the very simple sense that it is able to keep records of its own steps. This should not be considered a trivial matter: for instance, sometimes we are completely unable to describe how we perform certain actions or what the basic elements of certain skills are. The situation with Searle's reports, in the reply system response, is substantially the same. Searle may be conscious

of manipulating a certain symbol, i.e., the same symbol he used yesterday, even if he cannot be conscious *that* he is manipulating a certain symbol, that is, he does not know what the symbol means (2). The very fact that Searle may report his own activity on all this symbols' manipulation corresponds to being conscious *of*. One may argue that Searle's reporting activity actually is, again, symbols' manipulation, so that no level of consciousness could be reached through this activity. I disagree. When Searle reports his activity about symbols, he is using the symbols to refer to the symbols themselves. To be adequate to the task, Searle has to differentiate between an object-language and a metalanguage, and it is exactly this feature that defines the kind of consciousness I am discussing. In the case of Searle's native language understanding, on the other hand, the reports would be conscious reports in another sense. Specifically, Searle would be conscious *that* the content of the proposition he has in mind or has pronounced is such and such. The *that* clause gives to the report the intentional character Searle considers proper to the domain of real conscious understanding. In this way understanding, an intentional notion, is explained exclusively in terms of conscious understanding or, more specifically, of being conscious *that*. The problem here is that the distinction between being conscious *of* p and being conscious *that* p is not taken into consideration in Searle's use of the notion of consciousness. On the contrary, Searle's view seems to be committed exclusively with a notion of consciousness as a sort of "certainty" about what is going on in one's own mind, that is, only with the consciousness *that*. Considering the way in which we learn a language, this position could be disputed.

Suppose it is your first French class. The teacher tells you that "voiture" in French has the same meaning "car" has in English. As to your understanding, "voiture" was, up to five minutes earlier, a meaningless sound. It was exactly like a Chinese symbol for Searle. What differentiates you from Searle inside the Chinese room is that, in principle, you have direct access to the truth conditions for the correct use of "voiture." Why do you have this special privilege? Because you are at the right level for the use of a certain symbol: that is, you are at the causal interaction level between macro physical objects and audible sounds. What you have to do, and what you may do, is to point to a car, pronounce [vwaty:r] and wait for your teacher's reactions. In this sense you are in the same situation of Searle outside the Chinese room, i.e., you are in the same situation of Searle in his rebuttal of the systems reply. In this situation are you conscious *of* or conscious *that* ? Until your teacher confirms the correctness of your pointing, or you have reached a reliable basis of confirmation on the use of the sound, you seem to be conscious *of* saying [vwaty:r], not being conscious *that* you are saying "voiture." If this is correct, then learning may be characterized as the passage from being conscious *of* to being conscious *that* or, in semantical terms, from semantic performance to semantic competence, and not the other way round. If one accepts this conditional, then one cannot presuppose that being conscious *that* must always be the case; otherwise learning would be impossible. Since the process of learning could be described in the same way with respect to our first native language, where we have to correlate behavioral reactions and initially meaningless sounds, it is not possible to suppose that understanding is just a matter of being conscious *that*. So, if we are inclined to attribute forms of simple intentionality to kids, and perhaps to mute animals, we have to admit that being conscious *that* is not a requisite for intentionality, but that being conscious *of* is. We have, then, two different morals, a weak and a strong one. The weak moral says that since Searle does not differentiate between being conscious *of* and being conscious *that*, and does not take into account the being conscious *of* phase, he makes the process of learning more difficult to explain, leaving us without a clear idea on how we come to understand our first language. Therefore, while showing that -as beings capable of natural understanding- we are not computers, i.e., syntactical devices, the Chinese room argument fails to explain how we are capable of this natural understanding. The strong moral is the following: to argue against AI, Searle assumes consciousness *that*. Yet, as I have indicated, consciousness *that* requires consciousness *of*, this latter notion being perfectly "graspable" in strong AI. Since Searle does not take into consideration this distinction, he cannot have this latter notion. Consequently, without the consciousness *of* he cannot have the consciousness *that* either. If he cannot have this

latter notion, the Chinese room argument is no longer compelling.

Notes 1) I am simplifying a little bit in assuming that Searle already knows that the experiment is on Chinese. The most radical translation would be something of the form i) I am manipulating formal symbols for who-knows-what ii) I do not understand who-knows-what iii) Understanding is not manipulating formal symbols I think this further complexity may be avoided without losing anything in the argument 2) The distinction between being conscious *of* and *that* could be compared with that between awareness1 and awareness2 by Dennett (1969) or that between phenomenal consciousness and access consciousness by Block (1990, forthcoming).

Acknowledgments

Many thanks to Adriano Palma for helpful comments on a previous version of this paper. Thanks are due to Marc Mariani for checking my English and to an anonymous referee for this journal.

References

[1] Ned Block, "Consciousness Ignored. Review of Daniel Dennett Consciousness Explained", The Journal of Philosophy, 1990, pp. 181-193.

[2] Ned Block, "On a Confusion about a Function of Consciousness", to appear in Behavioural and Brain Sciences, 1995

[3] L.R. Carleton, "Programs, Language Understanding and Searle", Synthese, 59, 1984, pp. 219- 230.

[4] Daniel C. Dennett, Content and Consciousness, London: Routledge & Kegan Paul,1969.

[5] Daniel C. Dennett, "Fast Thinking" in The Intentional Stance, Cambridge Ma: Mit Press, 1987, pp. 323-338.

[6] John Haugeland, "Semantic Engines: an Introduction to Mind Design" in Mind Desing, Cambridge Ma: Mit Press, 1981, pp. 1-34.

[7] George Rey, "What's really going on in Searle's 'Chinese Room"', Philosophical Studies, 50, 1986, pp. 169-185.

[8] John R. Searle, "Minds, Brains and Programs", Behavioural and Brain Sciences, 3, 1980, pp. 417-424

[9] John R. Searle, Intentionality, Cambridge: Cambridge University Press, 1983.

[10] John R. Searle, "Is the Brain's Mind a Computer program?", Scientific American, 262, 1990, pp. 20-25.

[11] John R. Searle, The Rediscovery of the Mind, Cambridge Ma: Mit Press, 1992.

[12] Terry Winograd, "Understanding Natural Language", Cognitive Psychology, 1, 1972, pp. 1-191.

The Invitation to "Mar-Mur" Book

Witold Marciszewski and Roman Murawski:

MECHANISATION OF REASONING
IN A HISTORICAL PERSPECTIVE.

Poznan Studies in the Philosophy of the Sciences and the Humanities, Volume 43.

Editions Rodopi, Amsterdam-Atlanta, GA 1995.

ISSN 0303-8157, ISBN 90-5183-790-9, pp. 267.

1 About the Authors

Professor Witold Marciszewski is the head of the Department of Logic, Methodology, and Philosophy of Science at the University of Warsaw, Poland. Roman Murawski is a Professor in the Department of Mathematical Logic, Faculty of Mathematics and Computer Science, at Adam Mickiewicz University in Poznan, Poland. Both participated in the Polish research project concerning logical foundations of mechanized reasoning, supervised by W.Marciszewski.

Marciszewski writes on logic, logical philosophy, logic of language, the history of logic, etc., as in the recently published "Logic from a Rhetorical Point of View", Walter de Gruyter, 1994, and "Some Secrets of Internet" (in Polish), Warsaw 1995 (by the way, the latter is a wonderful book to encourage beginners in their first steps toward the Internet "know-how"). Murawski is widely appreciated for his books and papers on the history of mathematics and mathematical logic. MARciszewski and MURawski divided their text so that the authorship of the first chapters is attributed to the former, of the remaining ones to the latter. Still Marciszewski's perspective towards the problems discussed is present not only in "his own" part. On the other hand - the topics of chapters attributed to Murawski, in a sense define the scope of the earlier ones. So - truly speaking - one is bound to refer both names every time any one of them is mentioned. Instead, I shall refer to Marciszewski-Murawski's work as "Mar-Mur", this short form being applied to the authors' team.

2 General information on the book

Firstly: Who will profit reading it? - Students and researchers in Computer Science and AI willing to find a deeper philosophical and historical background for their professional activities; students and researchers in philosophy and humanities aiming at understanding - without tears - the links between Humanities and Mathematics, and between their Ideal Worlds and our common everyday technically determined environment. Historians of logic should be encouraged to see their subject in a new light, to wit as the HISTORY OF LOGIC IN THE PERSPECTIVE OF MECHANIZED REASONING. In this perspective, the three greatest steps in logic are seen as follows: (i) the formalistic approach (initiated by medieval nominalists and continued by Leibniz), (ii) algebraic formalism for a part of logic, which enables logical computing (from Leibniz to Boole), (iii) the reduction of the whole logic to that algebraic formalism, performed through elimination of quantifiers (Skolem, Hilbert, Gentzen, etc.). These three points form the main plot of the story.

Secondly: Who will like to read this book? - Everybody who is eager to KNOW of the World, who is also eager to "Know-What" about the early roots and philosophical motivation for "Know-How" in computing. The book is well written and there is no danger of feeling bored while reading it.

"There are as many stories as are "perspectives" into which we put data found in the sources" (p.11). The history of logic is no exception here. MarMur's perspective of tracing the early ideas in the history of logic is based on discovering the later interplay of "natural" reasoning against "artificial" reasoning. This artificial reasoning, again, is analysed as created in search of the nucleus (the best paradigm) in the natural reasoning. It is to be treated as the tool to extend and improve natural human mind's abilities of encoding and transmitting information.

Needless to say, talking today about that kind of stuff involves making use of a rather sophisticated conceptual apparatus. It also means the

danger of being involved in the endless discussions about this apparatus and opposing theories pretending to be the only legitimate users of this apparatus. The authors avoid that danger; as far as possible they take a "neutral" option. Due to this strategy the reader is able to follow what is the common base underlying possible differences. The reader will not encounter any inclination, e.g., to discuss "whether the logicist cognitive science is possible"; rather he/she will be confronted with what is cognitive science about and what are the contexts where logical theories must be recalled. No digression is made toward the Fodorian subtleties of the status of the human "internal language", but still there is put the problem of the nature (necessarily or not-necessarily linguistic) of the information encoded in the human mind (p. 33, items (1) - (3)). The authors' option is - so to speak - in agreement with a "sophisticated common sense". No special philosophical creed is needed to accept their points and follow arguments. One can find for instance that Mar-Mur's book ideas are independent both of the "early" ("functionalistic") as well as of the "later" ("anti-functionalistic") Hilary Putnam. In Marciszewski's line of commenting the subject matter, the simple, although quite often rather unknown, facts are recalled and collected together: they are the facts that speak for themselves.

The Authors seem to take for granted that once the model of Mechanisation of the language is formed, the remaining details are inessential. Is this perspective - one can guess - computer-style imitation of human brain's activity can succeed despite differences in the "nature" of the languages and the programmes of heuristic strategies.

"No one with a serious interest in the philosophy of mind or the philosophy of language can afford not to study it" - these are Stephen Schiffer's (from City University of New York) words about Hilary Putnam's book "Representation and Reality". One can repeat this opinion concerning Marciszewski-Murawski's book.

3 "Spacing"

There are seven Chapters, carefully divided into sections and subsections, fully listed in the "Extended Table of Contents" which closes the book. Together with instructive References (pp. 231-252), Index of Subjects and Index of Names, these devices friendly assist the search for information retrieval. Chapter 1 introduces the programme of the book. Some known definitions are recalled and some other nicely elaborated. For instance, the concept of Cybernetic Universe is introduced ("the world seen as consisting of information-processing machines") along with its suggested model-theoretical interpretation (p. 20). Among other topics in this chapter there are: the existence of model-based non-apperceived ("unconscious") reasonings; the Encoded Potential Concepts with which a machine should be equipped to match natural intelligence in model-based reasoning.

To some extent, this book can be defined by the names quoted in it. In Chapter 1 there are referred to, for example, the following authors: John von Neumann (1951, his theory of automata), Helmut Schnelle (1988, on "naturalisation" of logic), M. Davis (1988, on Post's contribution to computer science), Karl Popper (1982, the idea of a metaphysical research programme), H. Breger (1988, "know-how" in the context of mechanised reasoning), J. A. Makovsky (1988, the idea of Encoded Potential Concepts, referring to Chomsky's notion of linguistic competence), and R. Penrose (1988, 1989, mentioned when the Authors ask "at which level of complexity the codes in question are to be looked for", p. 43).

Chapters 2 - 5 deal with the roots of contemporary logical ideas adopted in the search on mechanisation of reasoning. (Chapters 2 and 3 by Marciszewski, the next two by Murawski). Chapter 2, "The Formalization of Arguments in the Middle Ages" is mainly about Ramon Lull whose legend is used as a background to show a real progress towards machanization of logic, due to the nominalistic school.

Chapter 3 discusses Leibniz's contribution to the idea of mechanisation of reasoning. Both Lull and Leibniz are Marciszewski's great fascinations. The first as (in a sense) "a black character" (I guess that one could add, to the Chapter 2, a motto: "Why Lull's name is still important, although he did not done the work he is praised to have done?"). Leibniz is of course praised as a "very VIP" in the history of logic. I should like to insist that the reader concentrates his attention on author's argument how strong was the call for

universal languages, ideography and algorithms in Leibnizian times. As I understand, the author suggests that there is a relatively straight way from Leibniz to contemporary cognitive science and AI (meanwhile - Goedelian results showed, why Leibnizian programme , after all, contained in itself some inevitably Utopian elements (p. 105)). Chapter 4 narrates what happened between Leibniz and Boole in the process tending to algebraization of logic.

Chapter 5 gives a careful and clear picture of the English Algebra of Logic in the 19th century. We find description of the logical machine of Jevons (now on display in the Oxford Museum of the History of Science), as well as information about other early logical machines. Many of the details included herein, even if well known to Anglo-American readers, give quite new information to continental readers, especially those not professionally occupied with the history of logic. In Chapters 6 ("The 20th century way to Formalization and Mechanisation") and 7 ("Mechanised Deduction Systems") new aspects appear. Historical motives are continued, but as we pass from Peano, Frege and Russell to Hilbert, Herbrand and Gentzen, the standard instruction on the topics in question is being included into the text. It results in the effect that one can rely on these chapters as on a well elaborated textbook. (In this context, see fragments in 6.8 on Gentzen's natural deduction and in 6.9 on Beth's Semantic Tableaux). "The History" commented in the last chapter includes results, projects and publications even of the last decade. The morning of this day seems to be treated as the history for the very same day's afternoon.

4 One particular question before saying "good-bye"

We read (p. 31) about formalised inference (data processing in the sphere of reasoning): "Though it has proved necessary for metamathematical research, as well as useful and insuring for philosophy of mind, it does not prove necessary for efficient reasoning. Albert Einstein could not do without arithmetical data-processing in his computations, but had no need to resort to rules of formalised deduction in his reasonings".

To illustrate this point a sentence is quoted which Einstein addressed (New York 1921) to the journalists who wanted him to explain briefly the basic tenet of his theory. He said: "If matter and its motion disappeared, there would no longer be any space or time". Nobody needs - Marciszewski comments - to learn the formalised record of the rule of transposition to acknowledge validity of inference of this statement from that one: "Matter and its motion results in time and space." It is linguistic competence together with "logical instinct" which do the work. However, I must confess that - after trying some small experimental job with students - I see some problems in the examples like this one: perhaps some knowledge of logical formalism is still welcome providing one is not identical with Einstein.

5 "Good-bye"

Let us take as inessential at which point the story about mechanised reasoning starts: with the Leibnizian "calculemus", or earlier? In the Middle Ages, or in Aristotle? Or has it started with Pythagorean expectations that the harmony expressed by numbers' proportions will tell us not only the reason why music means harmony while other sounds or noises do not, but even the secret of the harmony of all the universe? That is the way I would like to see it (which agrees with a small digression to Pythagor, see p. 28, f.9). Both moments taken as the starting point had something in common: it was the optimistic belief in the universal value of discovering the secrets of numbers. It was believed that through calculating information we could win in the Games with Nature. Despite that optimism I still wonder if we feel more happy profiting from the results of computer revolution. Caught in the NET, depressed by the explosion of information bomb, corrupted by the NETtian practices, can we dream of Harmony as the Ancients did? Are we the Governors of the NET, or are we nothing but little insects imprisoned in it?

CCAI Journal

Scope of the Journal

CCAI publishes articles and bookreviews relating to the evolving princi-ples and techniques of Artificial Intelligence as enriched by research in such fields as mathematics, linguistics, logic, epistemology, cognitive science and biology.

It is the intention to provide a forum for discussion of such topics as Cognitive modelling, logic programming, automatic learning, automatic knowledge extraction, AI and Art, applied epistemology and general aspects of AI.

Furthermore, CCAI is concerned with developments in the areas of hard- and software and their applications within AI. CCAI invites computer firms to submit special articles about new products, processes and/or information which they want to disseminate within the AI community as well as the business and industrial community.

Volume 12, 1995, number 1–2

Self-Reference in Cognitive and Biological Systems

Contents

L.M. Rocha: Introduction
H.H. Pattee: Evolving Self-reference: Matter, Symbols and Semantic Closure
R. Rosen: The Mind-Brain Problem and the Physics of Reductionism
C. Henry: Universal Grammar
L.M. Rocha: Artificial Semantically Closed Objects
G. Kampis: Computability, Self-Reference and Self-Amendment
P.R. Medina-Martins: Metalogues. An Abridge of a Genetic Psychology of Non-Natural Systems
P. Cariani: As if time really mattered: Temporal strategies for neural coding of sensory information

For more information please contact:
Communication & Cognition
Blandijnberg 2
9000 Gent (Belgium)
tel. +32 9 264.39.52
fax. +32 9 264.41.97

Gertrudis Van de Vijver
Senior Research Associate NFWO
University Ghent
Department of Philosophy and Moral Science
Lamoraal van Egmontstraat 18
B-9000 Ghent
tel. +32/9/222 07 28
fax. +32/9/220 73 05

Machine Learning List

The Machine Learning List is moderated. Contributions should be relevant to the scientific study of machine learning. Mail contributions to `ml@ics.uci.edu`. Mail requests to be added or deleted to `ml-request@ics.uci.edu`. Back issues may be FTP'd from `ics.uci.edu` in `pub/ml-list/V<X>/<N>` or N.Z where X and N are the volume and number of the issue; ID: anonymous PASSWORD: <your mail address> URL- `http://www.ics.uci.edu/AI/-ML/Machine-Learning.html`

THE MINISTRY OF SCIENCE AND TECHNOLOGY OF THE REPUBLIC OF SLOVENIA

Address: Slovenska 50, 61000 Ljubljana, Tel.: +386 61 1311 107, Fax: +386 61 1324 140.
WWW:http://www.mzt.si
Minister: Prof. Rado Bohinc, Ph.D.
State Secretary for Int. Coop.: Rado Genorio, Ph.D.
State Secretary for Sci. and Tech.: Ciril Baškovič
Secretary General: Franc Hudej, Ph.D.

The Ministry also includes:
The Standards and Metrology Institute of the Republic of Slovenia
Address: Kotnikova 6, 61000 Ljubljana, Tel.: +386 61 1312 322, Fax: +386 61 314 882., and
The Industrial Property Protection Office of the Republic of Slovenia
Address: Kotnikova 6, 61000 Ljubljana, Tel.: +386 61 1312 322, Fax: +386 61 318 983.

Scientific Research and Development Potential. The statistical data for 1993 showed that there were 180 research and development institutions in Slovenia. Altogether, they employed 10,400 people, of whom 4,900 were researchers and 3,900 expert or technical staff.

In the past ten years, the number of researchers has almost doubled: the number of Ph.D. graduates increased from 1,100 to 1,565, while the number of M.Sc.'s rose from 650 to 1,029. The "Young Researchers" (i.e. postgraduate students) program has greatly helped towards revitalizing research. The average age of researchers has been brought down to 40, with one-fifth of them being younger than 29.

The table below shows the distribution of researchers according to educational level and sectors (in 1993):

Sector	Ph.D.	M.Sc.
Business enterprises	51	196
Government	482	395
Private non-profit organizations	10	12
Higher education organizations	1022	426
Total	1,565	1,029

Financing Research and Development. Statistical estimates indicate that US$ 185 million (1,4% of GDP) was spent on research and development in Slovenia in 1993. More than half of this comes from public expenditure, mainly the state budget. In the last three years, R&D expenditure by business organizations has stagnated, a result of the current economic transition. This transition has led to the financial decline and increased insolvency of firms and companies. These cannot be replaced by the growing number of mainly small businesses. The shortfall was addressed by increased public-sector spending: its share of GDP nearly doubled from the mid-seventies to 0,86% in 1993.

Income of R&D organizations spent on R&D activities in 1993 (in million US$):

Sector	Total	Basic res.	App. res.	Exp. dev.
Business ent.	83,9	4,7	32,6	46,6
Government	58,4	16,1	21,5	20,8
Private non-p.	1,3	0,2	0,6	0,5
Higher edu.	40,9	24,2	8,7	8
Total	184,5	45,2	63,4	75,9

The policy of the Slovene Government is to increase the percentage intended for R&D in its budget. The Science and Technology Council of the Republic of Slovenia is preparing the draft of a national research program (NRP). The government will harmonize the NRP with its general development policy, and submit it first to the parliamentary Committee for Science, Technology and Development and after that to the parliament. The parliament approves the NRP each year, thus setting the basis for deciding the level of public support for R&D.

The Ministry of Science and Technology is mainly a government institution responsible for controlling expenditure of the R&D budget, in compliance with the NRP and the criteria provided by the Law on Research Activities. The Ministry finances research or co- finances development projects through public bidding, partially finances infrastructure research institutions (national institutes), while it directly finances management and top-level science.

The focal points of R&D policy in Slovenia are:

- maintaining the high level and quality of research activities,
- stimulating collaboration between research and industrial institutions,
- (co)financing and tax assistance for companies engaged in technical development and other applied research projects,
- research training and professional development of leading experts,
- close involvement in international research and development projects,
- establishing and operating facilities for the transfer of technology and experience.

JOŽEF STEFAN INSTITUTE

Jožef Stefan (1835-1893) was one of the most prominent physicists of the 19th century. Born to Slovene parents, he obtained his Ph.D. at Vienna University, where he was later Director of the Physics Institute, Vice-President of the Vienna Academy of Sciences and a member of several scientific institutions in Europe. Stefan explored many areas in hydrodynamics, optics, acoustics, electricity, magnetism and the kinetic theory of gases. Among other things, he originated the law that the total radiation from a black body is proportional to the 4th power of its absolute temperature, known as the Stefan-Boltzmann law.

The Jožef Stefan Institute (JSI) is the leading independent scientific research in Slovenia, covering a broad spectrum of fundamental and applied research in the fields of physics, chemistry and biochemistry, electronics and information science, nuclear science technology, energy research and environmental science.

The Jožef Stefan Institute (JSI) is a research organisation for pure and applied research in the natural sciences and technology. Both are closely interconnected in research departments composed of different task teams. Emphasis in basic research is given to the development and education of young scientists, while applied research and development serve for the transfer of advanced knowledge, contributing to the development of the national economy and society in general.

At present the Institute, with a total of about 700 staff, has 500 researchers, about 250 of whom are postgraduates, over 200 of whom have doctorates (Ph.D.), and around 150 of whom have permanent professorships or temporary teaching assignments at the Universities.

In view of its activities and status, the JSI plays the role of a national institute, complementing the role of the universities and bridging the gap between basic science and applications.

Research at the JSI includes the following major fields: physics; chemistry; electronics, informatics and computer sciences; biochemistry; ecology; reactor technology; applied mathematics. Most of the activities are more or less closely connected to information sciences, in particular computer sciences, artificial intelligence, language and speech technologies, computer-aided design, computer architectures, biocybernetics and robotics, computer automation and control, professional electronics, digital communications and networks, and applied mathematics.

The Institute is located in Ljubljana, the capital of the independent state of Slovenia (or S♡nia). The capital today is considered a crossroad between East, West and Mediterranean Europe, offering excellent productive capabilities and solid business opportunities, with strong international connections. Ljubljana is connected to important centers such as Prague, Budapest, Vienna, Zagreb, Milan, Rome, Monaco, Nice, Bern and Munich, all within a radius of 600 km.

In the last year on the site of the Jožef Stefan Institute, the Technology park "Ljubljana" has been proposed as part of the national strategy for technological development to foster synergies between research and industry, to promote joint ventures between university bodies, research institutes and innovative industry, to act as an incubator for high-tech initiatives and to accelerate the development cycle of innovative products.

At the present time, part of the Institute is being reorganized into several high-tech units supported by and connected within the Technology park at the Jožef Stefan Institute, established as the beginning of a regional Technology park "Ljubljana". The project is being developed at a particularly historical moment, characterized by the process of state reorganisation, privatisation and private initiative. The national Technology Park will take the form of a shareholding company and will host an independent venture-capital institution.

The promoters and operational entities of the project are the Republic of Slovenia, Ministry of Science and Technology and the Jožef Stefan Institute. The framework of the operation also includes the University of Ljubljana, the National Institute of Chemistry, the Institute for Electronics and Vacuum Technology and the Institute for Materials and Construction Research among others. In addition, the project is supported by the Ministry of Economic Relations and Development, the National Chamber of Economy and the City of Ljubljana.

Jožef Stefan Institute
Jamova 39, 61000 Ljubljana, Slovenia
Tel.:+386 61 1773 900, Fax.:+386 61 219 385
Tlx.:31 296 JOSTIN SI
WWW: http://www.ijs.si
E-mail: matjaz.gams@ijs.si
Contact person for the Park: Iztok Lesjak, M.Sc.
Public relations: Natalija Polenec

CONTENTS OF INFORMATICA, Volume 19 (1995) pp. 1–665

Articles

Agre, P.E.: *Computation and Embodied Agency,* Informatica **19** (1995) [4] 527–535.

Ahonen, J.J.: *Deep Knowledge and Domain Models,* Informatica **19** (1995) [2] 265–279.

Amoroso, R.L.: *The Extracellular Containment of Natural Intelligence: A New Direction for Strong AI,* Informatica **19** (1995) [4] 585–590.

Audsley, N.C., A. Burns, M.F. Richardson, and A.J. Wellings: *Data Consistency in Hard Real-Time Systems,* Informatica **19** (1995) [2] 223–234.

Bendix, L.: *Fundamental Tasks in Software Development Environments,* Informatica **19** (1995) [3] 391–405.

Bojadžiev, D.: *Gödel's Theorems for Minds and Computers,* Informatica **19** (1995) [4] 627–634.

Caplain, G.: *IsConsciousness a Computational Property?* Informatica **19** (1995) [4] 615–619.

Cherkauer, K.J.: *Stuffing Mind into Computer: Knowledge and Learning for Intelligent Systems,* Informatica **19** (1995) [4] 501–511.

Chrobot, S.: *Multi-Grain Rendezvous,* Informatica **19** (1995) [2] 241–255.

Colnarič, M. and D. Veber: *Supporting High Integrity and Behavioural Predictability of Hard Real-Time Systems,* Informatica **19** (1995) [1] 59–69.

Davari, S.: *On-line Algorithms for Allocating Periodic-time-critical Tasks on Multiprocessor Systems,* Informatica **19** (1995) [1] 83–96.

Dreyfus, H.L. and S.E. Dreyfus: *Making a Mind vs. Modeling the Brain: AI Back at a Branchpoint,* Informatica **19** (1995) [4] 425–441.

Erciyeş, K., Ö. Özkasap, and N. Aktaş: *A Semi-distributed Load Balancing Model for Parallel Real-Time Systems,* Informatica **19** (1995) [1] 97–109.

Gams, M., M. Paprzycki, and X. Wu: *Mind <> Computer: Introduction to the Special Issue,* Informatica **19** (1995) [4] 423–424.

Gams, M.: *Strong vs. Weak AI,* Informatica **19** (1995) [4] 479–493.

Goertzel, B.: *Artificial Selfhood: The Path to True Artificial Intelligence,* Informatica **19** (1995) [4] 469–477.

Gozzano, S.: *Consciousness and Understanding in the Chinese Room,* Informatica **19** (1995) [4] 653–656.

Hipwell, P.: *What Internal Languages Can't Do?* Informatica **19** (1995) [4] 647–652.

Katwijk, van J. and Hans Toetenel: *Loose Specification of Real Time Systems,* Informatica **19** (1995) [1] 25–42.

Leiss, E.L.: *Comparing Inference Control Mechanisms for Statistical Databases with Emphasis on Randomizing,* Informatica **19** (1995) [2] 257–264.

Li, X. and S. Bai: *An Algorithm for Self-Learning and Self-Completing Fuzzy Control Rules,* Informatica **19** (1995) [3] 301–312.

Lin, J., D.C. Kung, and P. Hsia: *An Object-oriented Approach for Modeling and Analysis of Safety-critical Real-Time Systems,* Informatica **19** (1995) [1] 43–58.

Lin, J.-F. and S.-J. Chen: *Performance Bounds on Scheduling Parallel Tasks with Setup Time on Hypercube Systems,* Informatica **19** (1995) [3] 313–318.

Liu, M.: *HLO: A Higher-Order Deductive Object-Oriented Database Language,* Informatica **19** (1995) [3] 319–331.

Maleković, M.: *A Sound and Complete Axiomatization of Functional Dependencies: A Formal System with only Two Inference Rules,* Informatica **19** (1995) [3] 407–408.

Marinoff, L.: *Has Turing Slain the Jabberwock?* Informatica **19** (1995) [4] 513–526.

McKay, C.W. and C. Atkinson: *Supporting the Evolution of Distributed Non-stop, Mission and*

Safety Critical Systems, Informatica **19** (1995) [1] 7–24.

Michie, D.: *"Strong AI": an Adolescent Disorder*, Informatica **19** (1995) [4] 461–468.

Orji, C.U. and T. Abdalla: *Performance Analysis of Disk Mirroring Techniques*, Informatica **19** (1995) [2] 209–222.

Paprzycki, M. and J. Zalewski: *Parallel and Distributed Real-Time Systems: Introduction to the Special Issue*, Informatica **19** (1995) [1] 3–6.

Paprzycki, M.: *Parallel Gaussian Elimination Algorithms on a Cray Y-MP*, Informatica **19** (1995) [2] 235–240.

Peschl, M.F.: *Methodological Considerations on Modeling Cognition and Designing Human-Computer Interfaces—an Investigation from the Perspective of Philosophy of Science and Epistemology*, Informatica **19** (1995) [4] 537–556.

Pissinou, N., V. Raghavan, and K. Vanapipat: *RIMM: A Reactive Integration Multidatabase Model*, Informatica **19** (1995) [2] 177–193.

Radovan, M.: *On the Computational Model of the Mind*, Informatica **19** (1995) [4] 635–645.

Runeson, P.: *Statistical Usage Testing for Software Reliability Control*, Informatica **19** (1995) [2] 195–207.

Schweizer, P.: *Cracks in the Computational Foundations*, Informatica **19** (1995) [4] 621–626.

Šlechta, J.: *On the Balance of the Informational Exchange, Its Flow, and Fractional Revealing Large Informational Quanta, in the 'Hot' Living Systems ($T < 0_-$)*, Informatica **19** (1995) [3] 333–344.

Souček, B.: *Quantum Intelligence, QI; Quantum Mind, QM*, Informatica **19** (1995) [4] 591–598.

Stufflebeam, R.S.: *Representations, Explanations, and PDP: Is Representation-Talk Really Necessary?* Informatica **19** (1995) [4] 599–612.

Tchouaffe, B. and J. Zalewski: *Fully Deterministic Real-Time Protocol for a CSMA/CD Type Local Area Network*, Informatica **19** (1995) [1] 123–132.

Walczak, S.: *Modeling Affect: The Next Step in Intelligent Computer Evolution*, Informatica **19** (1995) [4] 573–584.

Wasniowski, R.A.: *Nonlinear Adaptive Prediction Algorithm and Its Parallel Implementation*, Informatica **19** (1995) [3] 371–377.

Watt, S.: *A Brief Naive Psychology Manifesto*, Informatica **19** (1995) [4] 495–500.

Winograd, T.: *Thinking Machines: Can there Be? Are We?* Informatica **19** (1995) [4] 443–459.

Wojcik, Z.M. and Barbara E. Wojcik: *Optimal Algorithm for Real-Time Fault Tolerant Distributed Processing Using Checkpoints*, Informatica **19** (1995) [1] 111–122.

Wojcik, Z.M. and Barbara E. Wojcik: *Termination Conditions for Parallel Shape Recognition*, Informatica **19** (1995) [3] 379–389

Wu, J. and K. Yao: *Reliability Optimization of Concurrent Software with Redundancy*, Informatica **19** (1995) [3] 291–300.

Wu, X., S. Ramakrishnan, and H. Schmidt: *Knowledge Objects*, Informatica **19** (1995) [4] 557–571.

Yu, G. and L.R. Welch: *A Novel Approach to Off-line Scheduling in Real-Time Systems*, Informatica **19** (1995) [1] 71–82.

Železnikar, A.P.: *Principles of a Formal Structure of the Informational*, Informatica **19** (1995) [1] 133–158.

Železnikar, A.P.: *Elements of Metamathematical and Informational Calculus*, Informatica **19** (1995) [3] 345–370.

Profiles

Haneef Fatmi, Informatica **19** (1995) [1] 1–2.

Journal and Book Overviews

Železnikar, A.P.: *Minds and Machines*, Informatica **19** (1995) [1] 159–160.

Železnikar, A.P.: *Cybernetics & Human Knowing*, Informatica **19** (1995) [3] 409–414.

W. Marciszevski and R. Murawski: *Mechanisation of Reasoning in a Historical Perspective*, Informatica **19** (1995) [4] 657–659.

Calls for Papers

The Eight Australian Joint Conference on AI (AI '95). Informatica **19** (1995) [1] 161–163, [2] 281–283.

The First International Conference on Knowledge Discovery and Data Mining. Informatica **19** (1995) [1] 165–166.

International Conference on Software Quality (ICSQ '95). Informatica **19** (1995) [1] 167–168, [2] 286–287.

The Fourteenth International Conference on Object-oriented & Entity Relationship Modelling. Informatica **19** (1995) [1] 169–170.

Third International Conference on Computer Aided Engineering Education. Informatica **19** (1995) [1] 171–172.

Global Conference on Small & Medium Industry & Business (GLOCOSM). Informatica **19** (1995) [1] 173, [2] 288.

Electrotechnical and Computer Conference (ERK 95). Informatica **19** (1995) [2] 285.

Lukasiewicz in Dublin. Informatica **19** (1995) [2] 287.

Third International Workshop on Temporal Representation and Reasoning (TIME–96). Informatica **19** (1995) [3] 417–418.

First Workshop on Numerical Analysis and Applications. Informatica **19** (1995) [3] 419.

Conference Reports

Robič, B. : *EURO-PAR '95,* Informatica **19** (1995) [3] 415.

Šilc, J.: *ISCA '95,* Informatica **19** (1995) [3] 415.

Professional Societies

Jožef Stefan Institute, Informatica **19** (1995) 175, 290, 421, 661.

The Ministry of Science and Technology of the Republic of Slovenia, Informatica **19** (1995) 174, 289, 420, 660.

Errata

Because of the failure in the LaTeX font file `msbm` the letter $\mathbb{R}$ was printed deficiently (partly). This failure occurred in the paper A.P. Železnikar: *Elements of Metamathematical and Informational Calculus,* Informatica **19** (1995) No. 3, on the page 350, left column, lines 11 and 17. The correct text is the following:

To make the difference clear, let us introduce the operator $\mathbb{R}$ (for 'replace') instead of $\mathbb{S}$ (for 'substitute').

In a predicate formula $\mathfrak{A}$ (formula with predicates), a proposition A or a predicate $F(\cdots)$ can be replaced (substituted) by formula $\mathfrak{B}$. In this case, the substitution is marked by

$$\mathbb{R}_A(\mathfrak{A}) \quad \text{or} \quad \mathbb{R}_{F(\cdots)}^{(t_1,\cdots,t_n)}(\mathfrak{A})$$

respectively, where A is a variable proposition, F a variable predicate of n variables; formula $\mathfrak{B}(t_1,\cdots,t_n)$ includes among their free variables specially marked variables $t_1,\cdots,t_n$, the number of which is equal to the number of variables of predicate F, that is, n.

INFORMATICA

AN INTERNATIONAL JOURNAL OF COMPUTING AND INFORMATICS

INVITATION, COOPERATION

Submissions and Refereeing

Please submit three copies of the manuscript with good copies of the figures and photographs to one of the editors from the Editorial Board or to the Contact Person. At least two referees outside the author's country will examine it, and they are invited to make as many remarks as possible directly on the manuscript, from typing errors to global philosophical disagreements. The chosen editor will send the author copies with remarks. If the paper is accepted, the editor will also send copies to the Contact Person. The Executive Board will inform the author that the paper has been accepted, in which case it will be published within one year of receipt of e-mails with the text in Informatica LaTeX format and figures in `.eps` format. The original figures can also be sent on separate sheets. Style and examples of papers can be obtained by e-mail from the Contact Person or from FTP or WWW (see the last page of Informatica).

Opinions, news, calls for conferences, calls for papers, etc. should be sent directly to the Contact Person.

QUESTIONNAIRE

☐ Send Informatica free of charge

☐ Yes, we subscribe

Please, complete the order form and send it to Dr. Rudi Murn, Informatica, Institut Jožef Stefan, Jamova 39, 61111 Ljubljana, Slovenia.

Since 1977, Informatica has been a major Slovenian scientific journal of computing and informatics, including telecommunications, automation and other related areas. In its 16th year (more than two years ago) it became truly international, although it still remains connected to Central Europe. The basic aim of Informatica is to impose intellectual values (science, engineering) in a distributed organisation.

Informatica is a journal primarily covering the European computer science and informatics community - scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board.

Informatica is free of charge for major scientific, educational and governmental institutions. Others should subscribe (see the last page of Informatica).

ORDER FORM – INFORMATICA

Name: .....

Title and Profession (optional): .....

.....

Home Address and Telephone (optional): .....

.....

Office Address and Telephone (optional): .....

.....

E-mail Address (optional): .....

Signature and Date: .....

Referees:

Witold Abramowicz, David Abramson, Kenneth Aizawa, Alan Aliu, John Anderson, Catriel Beeri, Fevzi Belli, Istvan Berkeley, Azer Bestavros, Balaji Bharadwaj, Jacek Blazewicz, Laszlo Boeszoermenyi, Ivan Bratko, Jerzy Brzezinski, Marian Bubak, Leslie Burkholder, Frada Burstein, Wojciech Buszkowski, Ryszard Choras, David Cliff, Travis Craig, Wojciech Chybowski, Andrzej Ciepielewski, Tadeusz Czachorski, Sait Dogru, Georg Dorfner, Maciej Drozdowski, Mark Druzdzel, Hesham El-Rewini, Pierre Flener, Terrence Forgarty, Hugo de Garis, Eugeniusz Gatnar, James Geller, Janusz Gorski, Georg Gottlob, David Green, Herbert Groiss, Inman Harvey, Elke Hochmueller, Rod Howell, Ryszard Jakubowski, Piotr Jedrzejowicz, Eric Johnson, Li-Shan Kang, Roland Kaschek, Jan Kniat, Stavros Kokkotos, Kevin Korb, Gilad Koren, Henryk Krawczyk, Ben Kroese, Zbyszko Krolikowski, Benjamin Kuipers, Aarre Laakso, Phil Laplante, Bud Lawson, Ulrike Leopold-Wildburger, Joseph Y-T. Leung, Raymond Lister, Doug Locke, Andrzej Marciniak, Witold Marciszewski, Vladimir Marik, Jacek Martinek, Tomasz Maruszewski, Florian Matthes, Timothy Menzies, Dieter Merkl, Zbigniew Michalewicz, Roland Mittermeir, Madhav Moganti, Tadeusz Morzy, Daniel Mossé, John Mueller, Hari Narayanan, Jerzy Nogieć, Stefano Nolfi, Tadeusz Pankowski, Warren Persons, Gustav Pomberger, James Pomykalski, Gary Preckshot, Ke Qiu, Michael Quinn, Gerald Quirchmayer, Luc de Raedt, Ewaryst Rafajlowicz, Wolf Rauch, Peter Rechenberg, Felix Redmill, David Robertson, Bo Sanden, Guenter Schmidt, William Spears, Przemysław Stpiczyński, Maciej Stroinski, Tomasz Szmuc, Jurij Tasič, Piotr Teczynski, Ken Tindell, A Min Tjoa, Wieslaw Traczyk, Marek Tudruj, Andrzej Urbanski, Zyunt Vetulani, Olivier de Vel, Jo Weckert, Gerhard Widmer, Stefan Wrobel, Jusz Zalewski, Yanchun Zhang

EDITORIAL BOARDS, PUBLISHING COUNCIL

Informatica is a journal primarily covering the European computer science and informatics community; scientific and educational as well as technical, commercial and industrial. Its basic aim is to enhance communications between different European structures on the basis of equal rights and international refereeing. It publishes scientific papers accepted by at least two referees outside the author's country. In addition, it contains information about conferences, opinions, critical examinations of existing publications and news. Finally, major practical achievements and innovations in the computer and information industry are presented through commercial publications as well as through independent evaluations.

Editing and refereeing are distributed. Each editor from the Editorial Board can conduct the refereeing process by appointing two new referees or referees from the Board of Referees or Editorial Board. Referees should not be from the author's country. If new referees are appointed, their names will appear in the Refereeing Board. Each paper bears the name of the editor who appointed the referees. Each editor can propose new members for the Editorial Board or Board of Referees. Editors and referees inactive for a longer period can be automatically replaced. Changes in the Editorial Board and Board of Referees are confirmed by the Executive Editors.

The coordination necessary is made through the Executive Editors who examine the reviews, sort the accepted articles and maintain appropriate international distribution. The Executive Board is appointed by the Society Informatika. Informatica is partially supported by the Slovenian Ministry of Science and Technology.

Each author is guaranteed to receive the reviews of his article. When accepted, publication in Informatica is guaranteed in less than one year after the Executive Editors receive the corrected version of the article.

Volume 19 Number 4 November 1995 ISSN 0350-5596

Informatica

An International Journal of Computing and Informatics

Contents:

Introduction	Guest Editors	423
Making a Mind vs. Modeling the Brain: AI...	H.L. & S.E. Dreyfus	425
Thinking machines: Can there be? Are we?	T. Winograd	443
"Strong AI": an Adolescent Disorder	D. Michie	461
Artificial Selfhood: The Path to True AI	B. Goertzel	469
Strong vs. Weak AI	M. Gams	479
A Brief Naive Psychology Manifesto	S. Watt	495
Stuffing Mind into Computer: Knowledge and...	K.J. Cherkauer	501
Has Turing Slain the Jabberwock?	L. Marinoff	513
Computation and Embodied Agency	P.E. Agre	527
Methodological Considerations on Modeling...	M.F. Peschl	537
Knowledge Objects	X. Wu et al.	557
Modeling Affect: The Next Step in Intelligent...	S. Walczak	573
The Extracellular Containment of Natural...	R.L. Amoroso	585
Quantum Intelligence, QI; Quantum Mind, QM	B. Souček	591
Representations, explanations, and PDP: Is...	R.S. Stufflebeam	599
Is Consciousness a Computational Property?	G. Caplain	615
Cracks in the Computational Foundations	P. Schweizer	621
Gödel's Theorems for Minds and Computers	D. Bojadžiev	627
On the Computational Model of the Mind	M. Radovan	635
What Internal Languages Can't Do	P. Hipwell	647
Consciousness and Understanding in the...	S. Gozzano	653
Reports and Announcements		657