ON THE DESIGN OF DEVICES

WITH EMERGENT SEMANTIC FUNCTIONS

BY

PETER CARIANI

B.S., Massachusetts Institute of Technology, 1978
M.S. State University of New York at Binghamton, 1983

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Advanced Technology
in the Graduate School of the
State University of New York
at Binghamton
1989

© Copyright by Peter Cariani
All rights reserved

This is the electronic version of the thesis. Very minor, mostly typographical corrections have been made. Margins have been narrowed, so that the document is slightly more compact, and as a consequence, the pagination of this document differs a little from the original thesis.

This work was done in the Department of Systems Science, in the Watson School of Engineering. The State University of New York at Binghamton is now called Binghamton University, and the department is now called the Department of Systems Science and Industrial Engineering .

Comments, criticisms, thoughts, and suggestions are very much appreciated, and should be directed to:
Peter Cariani, 629 Watertown St., Newton, MA 02460. Cariani@mac.com; www.cariani.com.

Abstract

This dissertation examines the functional roles symbols play in biological organisms, scientific models and adaptive learning devices, analyzing the process of how symbols acquire new functions. The semiotic categories of syntactics, semantics, and pragmatics are used to examine the functioning of symbols in organisms, models, and devices. The dissertation explores how we would go about designing self-organizing devices which adaptively construct their own relationships to the physical world (emergent semantic functions). It bears on the frame/feature-generation problem in artificial intelligence, the problem of machine creativity in philosophy, the measurement problem in physics, the problem of generating new observables in science, and the problem of emergent functions in evolutionary biology.

An analytical framework is developed for distinguishing between computation, measurement, control, and nonsymbolic functionalities. A taxonomy of adaptive cybernetic devices is proposed in which nonadaptive devices ("formal-computational" and "formal-robotic") are distinguished from computationally adaptive ones ("adaptive devices", e.g. perceptrons, neural nets, genetic algorithms) and semantically adaptive ones ("evolutionary devices"). Evolutionary devices adaptively construct their own sensors and effectors contingent upon their performance, and are thus qualitatively different from devices now in use. Evolutionary devices are analogous to the biological evolution of new sense and effector organs. The capacities and limitations of the device types are outlined: evolutionary devices are capable of open-ended function generation, while the other device types must operate within closed sets of possible behaviors.

Three contemporary conceptions of emergence are distinguished: computational emergence, thermodynamic emergence, and emergence-relative-to-a-model. The levels of device adaptivity are connected to Rosen's theory of emergence relative to a model, where different types of devices diverge from their expected behavior in different ways.

Throughout, purely computational, logic-driven strategies for generating new functions (e.g. artificial intelligence, evolutionary simulation strategies, computationally-based artificial life) are criticized as being incapable of generating new primitives. An evolutionary robotics research program is suggested as an alternative.

Keywords: theoretical biology, biological cybernetics, emergence, cybernetics, self-organization, self-organizing systems, adaptive devices, adaptability, biological semiotics, artificial intelligence, systems theory, evolutionary simulations, artificial life, artificial evolution, theory of symbolic function, philosophy of biology, computers and philosophy

Dedication

to my parents, who would be proud and
to Becky, for her infinite patience, understanding, and love

Acknowledgments

Many people directly and indirectly have contributed to this dissertation.
I would especially like to thank: Howard Pattee, for his willingness to take the time and effort to contemplate and debate the conceptual issues, for his willingness to carefully read long, unwieldy, overly polemical manuscripts and edit them line-by-line in record time, and for his support through some very trying times, Eric Minch, for intensive debate and discussion of the issues, for his lengthy, incisive critique of the manuscript, for putting me up, as well as putting up with me, for staying my friend, James Pustejovsky, for giving an external perspective on the manuscript, for listening to my rants, for turning me on to William James, and for helping me make sense of Aristotle and Wittgenstein, Brent Cochran, for ongoing encouragement, for intellectual companionship, for small loans and old cars in times of financial adversity, for his sense of humor, Michael Kelly, for being the person he is, and for showing the rest of us the light at the end of the tunnel by finishing, Michael Hudak, for keeping me abreast of the computer bulletin boards and for giving me the opportunity to present my ideas outside the Pattee research group and to see once again what I am up against, Kevin Kreitman and Cliff Joslyn, for having a positive, optimistic vision of what the fields of cybernetics and systems science could be, Oliver Selfridge, Gordon Pask, Steve Heims, Jerry Lettvin, Chris Langton, Paul Rabin, and Rod Swenson, for useful, illuminating conversations and criticisms, Jonathan Jackson, for the generous use of his Macintosh for the final production push, Karen Cariani, for not asking me again when it would be finished, John LoManto, for giving me the courage to work on cars, for getting me to figure things out by taking them apart, and for helping me get them back together again when I couldn't do it myself, Past denizens of Fifth West, for their wild creativity and their can-do attitudes in the face of unrelenting pressure, Anne Wiseman, for her inspiring dedication to her students and her respect for us as people, for the intellectual explorations she catalyzed in all of us, the rest of my friends for their forbearance during my dissertation-induced absence.

I would like to thank all those in the Office for Sponsored Funds for their general helpfulness and the Graduate Office for their flexibility and trust regarding the manuscript guidelines.

This work was supported by a NASA Graduate Researchers Fellowship. I am grateful to those farsighted souls at NASA for making this highly interdisciplinary work possible.

Lastly, I would like to acknowledge the libraries and book stores of the Boston area, in particular the enlightened institution of open stacks at the MIT libraries and service beyond the call of duty by the librarians of the Connolly Branch of the Boston Public Library.

Contents

(NB: page numbers are from the original thesis; additional page numbers are for this version)

Preface

> *Mental facts cannot be properly studied apart from the physical environment of which they take cognizance*....our inner faculties are *adapted* in advance to the features of the world in which we dwell, adapted, I mean, so as to secure our safety and prosperity in its midst....Mind and world in short have evolved together, and in consequence are something of a mutual fit....*mental life is primarily teleological;* that is to say that our various ways of feeling and thinking have grown to be what they are because of their utility in shaping our *reactions* on the outer world....the essence of mental life and bodily life are one, namely "the adjustment of inner to outer relations." (William James, 1892, pp.xxvii-xxviii).

The evolved nature of the relationships between mind and world, which seemed so clear to William James a hundred years ago, has today become buried beneath a mass of postulated logical mechanisms, ones in which the connection of symbols to the outside world has been reduced to a collection of truth values. The *functions* of symbols, whether utilized by minds, organisms, or devices are overshadowed by the mechanisms for their manipulation. And if questions of function are widely obscured, what of questions involving the *origins of new functions?*

The origins of semantic functions, the relations between symbols and the external world, tend to be such neglected questions. In science this involves how new observables come into being. In biology this involves how sensory apparatuses of organisms evolve over time. In robotics it involves how new primitive features and actions can arise through adaptive self-construction. The difficult analysis of the process of creating new observables, in evolution, in science, in the design of learning devices, has been passed over in favor of more tractable problems of uncovering the logical relations between symbols. In evolutionary theory this has meant a predominance of analysis of gene frequencies over considerations of phenotypic function. In the philosophy of science it has meant a predominance of analysis of the formal aspects of scientific models rather than considerations involving the relations of the models to the material world and their usefulness to us. In machine learning it has meant the search for purely computational solutions independent of any real world mappings. Consequently, the question of how symbols acquire semantics has been systematically subsumed into questions of syntax, how symbols relate to each other. And just as systematically, questions of pragmatics, why symbols come to have the semantic and syntactic relations they do, have been left out of the picture entirely.

A biological semiotics can help us to recover these lost dimensions of symbolic function and to uncover the evolutionary mechanisms by which biological organisms come to acquire new relations to their world. Over evolutionary time periods and over all levels of organization, from enzymes to tool-making, organisms acquire new capabilities for sensing and acting; they implement new semantic functions. How would machines do this? *We need such a theory of biological semiotics if we are to build devices which construct their own semantic relations to the world.* I believe that such a capability is necessary if we are to break out of the trap of always having to completely define the primitive terms that our devices operate with. We will need devices firmly embedded in the real world which construct their own semantic relations, their own primitive features and actions, their own sensors and effectors. If we want them to be creative, thereby enriching and enlarging our own semantic repertoires through their operation, we must give them the structural autonomy necessary for transcending our specifications. When this happens, our devices will have emergent properties relative to us, functions not reducible to what we already know. Our devices will afford us a means of enlarging the basic observables of our world.

Most of this dissertation is devoted to building up a theory of the semiotics of autonomous organisms and showing how it could be applied to the analysis of constructible devices. But it is also necessary, simultaneously, to criticize the dominant computationalist assumptions which have effectively marginalized questions of semantics and pragmatics. As long as these assumptions stand unquestioned, there is no room for an alternative view. Consequently, many deeply embedded tacit

idealizations and conventional truths of artificial intelligence, computer science and artificial life must be challenged before any alternatives can be seriously contemplated.

Currently there is a battle raging in cognitive science between top-down "classical" symbolic manipulations and bottom-up, parallel distributed processes (PDP). While the biological, massively parallel organization of brain function has been integrated into neural net and connectionist models, neither camp ever directly addresses the question of the origin of *external* semantic primitives. As Margaret Boden has observed, the "radical departure of connectionism" is really still "a change in some relatively basic ideas *within* the general computationalist approach" (Boden, 1988, p.252). Even the more biologically-minded connectionists have overlooked a central feature of cognitive evolution, that the nervous systems of higher organisms co-evolved with the sense organs and effector organs they coordinate. If computation is the abstract analogue of neural coordination, then measurement and control are the analogues of biological perception and action. There has been far too little thought about how these latter, *semantic* functions might evolve. Just as an associative memory, whatever its other advantages, can't compete with direct addressing for efficiency, it seems doubtful to me that the connectionists can effectively compete with the classical symbol processing for carrying out arbitrarily chosen logical operations. Classical symbol processing seems to be the paradigm best adapted for completely syntactic operations. On the other hand, connectionist systems might be better suited to realizing semantic operations, in finding ways to connect the symbols of the classical systems to the world. What would be needed is a reconceptualization of the *purpose* of connectionist devices, from devices which clumsily carry out logic-based, syntactic computations to those which implement what no classical symbol system can do, to sense and to act, and to do these things adaptively.

What would an alternative research program look like which incorporated noncomputational, semantic processes of measurement and control, and dealt with their origins? It would involve a kind of evolutionary robotics in which measurements and actions were every bit as important as computations. It would involve a pragmatist perspective where the device is an integrated whole operating in the real world to perform real tasks. We would in effect co-evolve the sense receptors, computational coordinations, and effectors our devices needed for a specific task. This evolution would involve pragmatic relations as well as syntactic and semantic ones: real world functioning, evaluation of performance, and selection among alternative structures. We would have to think long and hard about what kinds of constructive systems would supply the analogues of transcription, translation, and protein folding in organisms, but there is no reason to doubt that at least some chemical or electronic analogues could be found. It would also behoove us to deeply consider early on the likely social consequences, both good and bad, of such a qualitatively new technology.

Personal evolution of the problem

As a student of biology, I was most concerned with issues of biological organization and structural change: questions regarding morphogenesis, aging, and evolution of complex morphologies. I have always been most fascinated and puzzled by the apparent explosion of biological diversity and complexity, the evolution of qualitatively new functions. How does it all happen at once? For me, the overall increase in the functional complexity of living things over phylogeny, what von Neumann called the "principle of complication," is the central feature of biological evolution. Structures and functions which are unimaginable from earlier evolutionary vantage points appear in an apparently open-ended way. It is difficult to imagine how one would go about even circumscribing *ranges* of functionalities which are available to evolving populations of biological organisms. Maybe it would be possible to generate qualitatively new functionalities through simulations of morphological evolution. This I was determined to try to do.

I came to my current position, that computer simulations are incapable of generating this kind of open-ended diversity, only after attempting to devise evolutionary simulations with open-ended,

emergent behaviors. After attempting a few of these and many gedanken-simulations, I thought it was obvious that *no* simulation would be capable of fundamental novelty; a simulation only reflects the assumptions programmed into it from the start, yielding only the logical consequences of those assumptions. Each simulation operates within a closed world, one bounded by its defining notation. Where is the exit out of *this* notational trap? It must lie in structures having some properties which are independent of the programmer. In short, *biological organisms are autonomous relative to us, while computer simulations are not.*

I soon found out that these ideas were not at all obvious to everybody else. Many simply denied that simulations were closed, pointing to infinite computations or fractals or chaos; others would not even recognize the problem of fundamental novelty, claiming that all evolutionary developments were deducible from previous ones. I found these responses to my simple observation inadequate, either evading the question entirely or assuming, like Turing (1950), that there would always be a programmer there to enlarge the simulation to take care of anything that might have been left out. On the other hand, the critiques of Dreyfus (1979) and Searle (1980), which influenced my early thinking, did not seem entirely satisfactory. Coming out of the cognitively impenetrable realms of Husserl and Heidegger, their critiques don't explicitly point the way to an alternative research program. Unlike the American pragmatists, whom I discovered later, their perspectives, involving transcendental conceptions of intentionality and consciousness, don't mesh easily with a concrete, biologically-minded, evolutionary worldview.

Once simulations are seen to be closed, while biological evolution is seen as open ended, one can ask what the nature of the difference between biological organisms and computers is, such that biological organisms evolve in an open-ended way, while computer programs do not. I feel the answer lies in the nature of biological symbols and how they are tied to the world at large versus their computer counterparts. Unlike the symbols manipulated in computer simulations, biological symbols are connected to the world through the biological structures they *construct* and they are selected on the basis of their manifold real world effects. As a consequence, biological organisms possess an adaptivity arising from the way that they are connected to their environments that is not found in computer simulations.

In populations of biological organisms, new functions not reducible to combinations of old ones can emerge precisely because of the structural plasticity of biological organisms. The places where this is most apparent are in the evolution of new detectors, from receptor molecules to sense organs, permitting fundamentally new sensitivities, and new effectors, body structures, permitting fundamentally new actions. I began to think about different levels of adaptivity, and especially what kinds of adaptivity are necessary for emergent functional novelty.

Comparing adaptivity in biological organisms and trainable computing machines, I could see the rough equivalents between microevolution and trainable classifiers. Each is optimizing within a fixed set of possibilities, be it a pool of genetic alleles or combinations of weighting parameters. But there was no equivalent for morphological evolution on a phylogenetic scale for computing devices. I saw the difference here being the physical self-construction of biological organisms versus the fixed structure of the computing device.

Physical construction is crucial. The organism or device must have some degree of structural autonomy in order to do anything fundamentally unexpected, to defy the notation. A simulated constructed organism would still be subject to the constraints of simulations. There would be a construction language within which all the simulated possible structures would lie. I concluded that to have real open-endedness of possibility, *we cannot merely simulate, we must build real devices* having some physical properties which we have not prespecified in advance. *If we want to get emergent real world functions, the device must itself operate in the real world and not in some simulated symbolic realm.* Like the observation regarding the closure of simulations, this conclusion has been difficult for

many to accept. We argued this point in our research group for several years while we were at the same time laboriously trying to clarify for ourselves the concepts of symbol, computation, measurement, and control. Even defining the primitives of our analysis has been an enormously difficult undertaking. If the reader has any doubts, I suggest that s/he try to explain exactly what a "symbol" or a "computation," or, better yet, a "measurement" is to someone who takes these meanings for granted.

It took quite a long time for our ideas regarding these analytical primitives to solidify, but once one has a clear idea of what constitutes a symbol and how symbolic behavior can be recognized in nature, many general similarities between biological organisms and devices appear. Considering various types of devices and organisms which utilize symbols, it is possible to decompose the various relations into completely symbolic, deterministic transformations (computations) and mixed symbolic-nonsymbolic, contingent transformations (measurements, controls). Further, these transformations can be combined in various ways to yield different types of symbol-utilizing devices. Contemporary devices can be classified in these terms.

The behavior of devices which only perform computations can be completely simulated. From this I concluded that any device which exhibits fundamental emergence must be adaptive in other parts of the device, namely in the measurements and controls. Put another way, whatever emergent potential exists must lie outside of purely digital processes in the realm of analog and mixed digital-analog processes. One sees the barest hints of this perspective in von Neumann's *The Computer and the Brain* (see appendix 5). For analog processes, there are always interactions going on that will be outside the formal model or simulation. The central purpose here, though, is not to simply generate pragmatically meaningless fluctuation-sensitive sensors and effectors, but to generate new structures with some usefulness to the organism or device. *Random, stochastic, or chaotic physical processes unconnected to improving or creating functions will not do us any good*. I have always thought that concepts of self-organization and emergence are crucial--they played a large role in *my* formation--but, so much of the discussion of emergent physical structures simply never connects these newly-formed structures to functions of any kind.

How would we do this in a device? It took a long time for me to see how this might be done. First, we would need to make the structures *variably constrainable*, so that we could freeze out degrees of freedom, thereby preserving the structure when we found one with a useful function. We would need a means of controlling the level of structural variability, like the thumbscrews on a drafting lamp or the "temperature" control of a simulated annealing process. Second, we would need to *make the constraint of these structures performance-dependent*. The thumbscrews would be loosened or the temperature would be increased when we wanted to search for new, better configurations and then these constraints would be tightened down once we found something we liked. I believe something like this goes on in protein folding, via the selection-driven substitutions of amino acids in the chain having greater or lesser flexibility according to whether evolutionary pressures demand innovation or efficiency.

Given that a mixed digital-analog device having the property of performance-dependent construction of sensors and effectors is necessary for the kind of emergent behavior I had in mind, I went searching the literature for various kinds of adaptive devices that had been envisioned. Had anyone actually built a device with emergent semantics, one which constructed its own sensors and effectors? Had anyone even discussed such a possibility? There is surprisingly little mention of this, largely, I think, because very few people distinguish measurements and controls from computations. The distinction gets obliterated completely in all the standard treatments, to the extent that computer simulations and robotic realizations are often not separated. Biological systems are seen only as enormously larger, massively-parallel computing systems with orders of magnitude more computational power than contemporary computers. I see a pervasive tendency in artificial intelligence and computer science to believe that literally *everything* is a computation. If, consequently all devices are really computers, how can one even conceive of devices which do more than compute? Seen this way, such devices would violate Church's

Thesis in its strong form, which is often held as an article of faith. By the time this faith reaches its culmination in the ideas of Fredkin and Toffoli, it knows no bounds:

> In a sense, nature has been continually computing the 'next state' of the universe for billions of years; all we have to do--and, actually, all we *can* do--is 'hitch a ride' on this huge ongoing computation, and try to discover which parts of it happen to go where we want. (Toffoli, 1982, p.165).

Amidst many Laplacian pronouncements like this, which seem completely unconnected to how anyone would ever empirically test this claim, I have tried as much as possible to ground the *descriptive* concepts of computation, measurement and control in directly observable differences of behavior. Beyond this attempt to separate out what we would want to label in our encounters with nature as a computation, it was also necessary for me to recover *functional or pragmatic* conceptions of computations, measurements, and controls: What do they do for us? When do we want computations? Why do we make measurements? What do controls accomplish? These are, of course, questions of engineering, of practical design. All engineers intuitively understand the distinctions between A-to-D sensors, digital logic-devices, and D-to-A effectors *as ones involving different purposes*. I believe a major reason the devices I sought in the literature are not there, even in theory or speculation, has to do with the separation of computer science from general engineering practice, of software from hardware. Perhaps the expanding field of robotics will ameliorate this situation.

The best that contemporary robotics seems to offer is the prospect of "tunable" sensors and effectors, and these efforts are exceedingly important, but again, they amount to optimizing a structural parameter, an existing function, not creating a qualitatively new observable, a new function. As far as I can see from the literature, only one person, Gordon Pask, has directly dealt with the problem of devices adaptively generating new observables, in his papers of the late 1950's dealing with organic analogues to the growth of a concept. At a cybernetics conference in 1984 I had heard Stafford Beer talk about the spontaneous emergence of an "artificial ear" from a vat of electrolytic salts. Much later, while looking for some other papers, I came across Pask's paper in the 1958 conference proceedings on *The Mechanization of Thought Processes*. I do not yet know what became of this research direction then, but his experiments seem to be an appropriate place to start the alternative research program I have in mind. In the near future I will try to replicate his experiments.

Intellectual debts

To Howard Pattee I owe the deepest intellectual debt. As an undergraduate alienated from a biology department so molecularly-minded that no course on evolution or ecology was deemed worth offering, I scoured the libraries for alternatives. Luckily, I came across the series *Towards a Theoretical Biology* edited by C.H. Waddington and *Biogenesis, Evolution, Homeostasis*, edited by Alfred Locker. In the latter book, Pattee and Robert Rosen presented papers dealing with the origins of new biological structures and functions and the inherent limitations of existing mathematical models to deal with the problem. These two papers (Pattee, 1973; Rosen, 1973) had a profound effect on my thinking: emergence would either involve a new kind of mathematics or it would have to happen outside of simulations entirely. Aside from Pattee and Rosen, virtually no one else seems to have addressed the problem of biological novelty on this fundamental level.

Pattee has developed a conceptual framework which addresses the problem of symbols in biological systems from the point of view of physics and has argued that there are irreducible, complementary linguistic and dynamical descriptions of these systems. The experience of quantum mechanics taught the necessity for taking the measuring devices we utilize into the account when we describe our theories and has reminded us of the irreducibility of measurements and computations. Symbols are, of course, intimately bound up in the measurement instruments and the mathematical calculations. The molecular biology revolution brought the startling realization that *virtually all biological organisms involve the use*

of symbols at their very core. In physics and in biology, symbols come to play a pivotal role. Early on, Pattee (1968) analyzed symbolic activity in terms of special types of physical constraints, "nonholonomic" ones, contrasting the organization and functioning of symbols in machines and organisms. From the complementarity between the various alternative descriptions of biological systems, he developed the concept of *semantic closure* and the necessity for the reliable construction of measuring devices by the biological organism. It will be apparent that almost all the basic categories of my analysis can be found in one form or another in Pattee's conceptual scheme. If one sees enzymes as making measurements, and cells adaptively selecting which enzymes they will construct, then all of the relations for what I call an evolutionary device are present. All that remains is the articulation of this set of functional relations into a new kind of humanly constructible artefact.

The hammering out of the basic concepts, their debate and clarification through interminable discussions, would have been difficult, if not impossible, without the group of graduate students who were familiar with the basic questions and the work of Pattee and Rosen: Eric Minch, Michael Kelly, Dennis Waters, Michael Hudak, Eileen Way, and Nicholas Demeris. In particular Eric Minch and I have had numerous debates over what types of emergence can take place in computer simulations. His work (Minch, 1988), which involves the formation of new "objects" in simulated reaction networks, involves how one would go about simulating emergent processes on a computer. Whether these resulting computational structures are truly "emergent" and whether they can have genuine functions (e.g. make real measurements) are issues which will requiring ongoing re-evaluation on both of our parts as our respective researches progress.

Although I have never worked with him, Robert Rosen's ideas regarding the relationship between formal models and natural systems have also been an important source of ideas. Rosen most directly links theoretical biology with systems theory, grappling with the capacities and limitations of formal models. In the process he has rigorized the concept of emergence by making it relative to a formal model.

Protein-folding can be seen as a primary example of a biological construction system. My work with Narendra Goel on computer simulations of protein folding (Cariani & Goel, 1985) stimulated a whole line of thought regarding what made this process effectively open-ended in function. Here the necessity for stable folding intermediates seems to indicate at least some hierarchical organization of the folding process. This structural hierarchy is a result of a mixture of more constrained segments (secondary structures) and less constrained ones (turns). The chain can thus implement varying degrees of constraint according to the flexibility of the various segments which is dependent upon their constituent amino acids. The evolution of highly developed enzymatic function can be seen as the optimization of the degree of constraint of the chain in its folding. For devices which perform constructions this feature is a very desirable, and perhaps even a necessary, design feature.

On another front Goel was beginning to deal with the problem of correctly classifying crops on the ground from satellites situated in space, but it was not clear at first if the LANDSAT measurements (i.e. set of observables) contained enough information to effect a useful classification. If not, adaptive, tunable sensors might be something to contemplate. We encountered a similar situation in protein folding: which aspects of the protein must be represented to effectively make even a rough, global prediction of the folded state? Are electrostatic interactions crucial? What about sidechain stereochemistry? How precisely does the molecular geometry need to be represented? The choice of effective observables was (and still is) far from obvious for this difficult problem.

Many of my ideas concerning machine learning came out of a course in "adaptive programming" taught by Don Gause. The wide range of examples, from adaptive conversation machines to genetic algorithm graphics generators, emphasized real world performances and feedback. In his class I realized that every trainable classifier is limited by its primitive features and that another level of adaptivity is possible, one in which the primitive features themselves are selected. His strategy of taxonomizing

adaptive devices directly inspired my taxonomy (Gause & Rogers, 1982); his "heuristic problemsolving" techniques all involve transcending fixed interpretational frames by searching through an undefined, subconsciously held space of possibilities.

The First Artificial Life Workshop at Los Alamos in September, 1987 was also an important event. The conference afforded a bird's eye view of the state of theoretical biology. Although the concept of emergence was recognized as a problem, there was no real debate over what the concept means or whether there is a qualitative difference between the organism and its computer simulation. I felt these basic questions should have been at the core of artificial life, but they were hardly addressed at all. The most disappointing aspect was the degree to which computationalist assumptions have dominated the definition of this embryonic field (e.g. Langton, 1989). More than anything else, the lack of an alternative view within artificial life has convinced me of the importance of articulating one, and thus has propelled me into writing this dissertation. Artificial life must not replicate the profound philosophical errors of its predecessor, artificial intelligence.

My ideas have, of course, been influenced by other theoretical biologists: Ludwig von Bertalanffy, C.H. Waddington, Paul Weiss, D'Arcy Thompson, Humberto Maturana, Rupert Riedl, and Stephen Jay Gould. As an undergraduate I was ingrained with a deep respect for biological functionality through the critical skepticism, the feisty iconoclasm, the conceptual articulateness, and the good humor of Jerry Lettvin. All of these scientists point in one way or another to the development of a coherent theory of biological organization, one going beyond the enormous compendium of biochemical mechanisms that passes for biology today.

From the beginning cybernetics and systems theory have been intertwined with theoretical biology. In cybernetics many of the ideas of biological regulation, adaptation, and evolution were first applied to the design of learning devices and human organizations. Margaroh Maruyama's epistemological taxonomies (and "deviation-amplifying" processes) as well as Heinz von Foerster's notions of self-organization shaped much of my formative thinking concerning the problem of generating open-ended behavior. To John von Neumann we all owe much clarity of thought concerning the role of symbols in self-reproducing automata, the necessity for mixed digital-analog processes, the limitations of simulations, as well as a simple, coherent framework for discussing the basic functions of the brain. Through ongoing exposure to George Klir's general system's methodology, I came to see 1) that a frame of observables could be rigorously defined, 2) that the resulting "appearances" themselves could be analyzed and 3) that all deterministic formal models could be taxonomized in a systematic way. I came to the texts of W. Ross Ashby relatively recently, but many of his ideas regarding types of adaptivity run parallel to mine, and it is hard to shake the feeling that these ideas, all latent within cybernetics and general systems theory for a good thirty years, are finally coming back in a new form.

Cybernetics and systems theory have sometimes, inexplicably, seemed to be at odds with each other. Cybernetics traditionally has been concerned with analysis of structural relations and construction of control mechanisms, while systems theory has been involved with the identification of underlying mechanisms from their observed behavior. One focuses on mechanisms, the other on models. Both approaches are necessary. Without *some* prior understanding of mechanisms and relations, systems theory becomes a futile, unconstrained search through a model space with only the bare empirical data as a guide. Without *some* means of identifying types of mechanisms from their observed behaviors, cybernetics cannot extend its analysis from human artefacts to the mechanisms and relations of the natural world. It is therefore imperative to analyze both the structure of their mechanisms and their "structure of appearance." Most of this dissertation concentrates on the cybernetics side, analyzing various kinds of feedback loops and the functions they enable with the construction of actual devices in mind. The systems theoretic side is embodied in the analysis of observed state transition structures (chapter 5) and in the conception of emergence relative to a model (chapter 12).

This dissertation must necessarily address some of the classical debates which have been raging in natural philosophy for over two millenia. As C. West Churchman so elegantly demonstrated in his *Design of Inquiring Systems*, every philosophical system is a prototype of a cognitive device and every cognitive device implements a particular philosophy of mind. Every philosophy of mind is indirectly a philosophy of science and every philosophy of science is a general problem-solving strategy.

The general philosophical perspective of this dissertation lies very close to that of the American pragmatists, particularly that of William James and John Dewey. This should not be surprising, since pragmatism is a product of the Darwinian revolution. I came to the pragmatist, behavioral linguistics of Charles Morris late in this project, but his categories seem to be very powerful unifying principles. There are deep interconnections between general theories of symbol function and pragmatist philosophies of science. This connection between symbols and the world is expressed in the concrete clarity and connectedness to real world observation in the epistemologies of Hermann von Helmholtz, Heinrich Hertz, Ernst Mach, and Niels Bohr. I take their insistence on observability to imply not that our worldview should be limited to what we can observe, but that the conditions for reliably using, testing, and communicating our models are different than those needed for creating new models. This is basically like saying that the process of genetic variation and the construction of alternative phenotypes is different from the process of performance and selection. As Poincaré noted, imagination and creativity lie outside the formal symbolic manipulations of the physical models we use to make predictions; the creativity of the physicist and the mathematician involve the creation of new interpretations for symbols, the semantic part, rather than finding new ways to push around old tokens. The systematic critique of realism launched by Richard Rorty (1979, 1985), the "epistemological anarchism" of Paul Feyerabend (1973, 1978, 1981a, 1981b, 1987), and the "constructive empiricism" of Bas van Fraassen (1980, 1985) only seem coherently linked when the realist criterion of truth is replaced by a pragmatist criterion of usefulness relative to a particular context. Relativism will not appear to be a sensible alternative unless it is seen in this way. I would argue that incommensurability between theories stems from both different semantic relations (different sets of observables) *and* different pragmatic demands arising from different contexts (different evaluative criteria). It is only in the syntactic realm, all other relations being the same, that strict logical comparisons can ever be made. In my opinion, if we see these issues in comparative biological terms, it doesn't make sense to say that a seagull has a more realistic model of the world than a lobster, nor does it make sense to argue whether a domesticated animal has a more effective model of the world than its wild counterpart, since they face radically different challenges. Various models with different sets of observables and purposes are as different organisms (see Munévar, 1981 for the implications of this view for the philosophy of science).

I was drawn into the philosophy of mathematics partly as a result of the responses I was getting from asserting the closed nature of models and simulations. These ranged from questions about whether stochastic state transitions, meta-rules, mathematical chaos, fractals, or connectionist and neural net "front ends" might somehow alter the situation to various postulations involving infinite operations on infinite strings, virtual worlds, and other platonic fantasies. Fifteen years ago it would have been hierarchy-formation and catastrophe theory. These are all interesting and useful theories, but, unfortunately, they don't alter the bounded nature of computer models and simulations. Much of the confusion has been caused by the lack of a clear distinction between *simulations* and *implementations*, between *representations* and *realizations*. Gödelian computability arguments, which I believe are irrelevant for physically realizable, finite devices, are often brought up as if they will magically resolve the debate by guaranteeing formal undecidability. In my opinion they do not (see appendix 1). Some of these responses were exceedingly hostile, and I regret to say that they have had noticeable effects on my presentation. If some of my points seem a bit nitpicking or arcane, or if I tediously draw out my definitions, this is why.

To insist on the observability of mathematically postulated entities runs against the grain of mathematics, which tends to be very platonic in its outlook. All along, however, there have been those who have insisted on at least some concrete relation between mathematical objects and the material world. I count Aristotle, Kronecker, Poincaré, Hilbert, many of the intuitionists, the later Wittgenstein, and the strict finitist-constructivists (appendix 1) as loosely belonging to this camp. As of late I have come across the existence of the "ultra-finitist" position of Esenin-Vol'pin, which apparently carries these ideas further, challenging the conventional assumption of a unique sequence of natural numbers outside the direct grasp of our symbol manipulating instruments (Troelstra & van Dalen, 1988, chapter 1).

In my mind there are strong conceptual connections between mathematical constructivism, Bohr's epistemology, biological evolutionism, Piaget's evolutionary constructivism, "language as cooperation" and pragmatist philosophy. Neither time nor space permits an extended, comprehensive discussion of their adversaries: Platonic, rationalist, realist, and mechanistic reductionist viewpoints, which are also all related in various ways. As I understand them, the metaphysical premises of these worldviews are, in a very deep way, at odds with any account of fundamentally emergent structures and functions.

Lastly, history and social theory have also shaped my thinking. Some of my earliest thoughts about structural change were formed in high school while reading about cyclic and evolutionary theories of world-history: Toynbee, Spengler, Kroeber, Sorokin, Kondratieff, Parsons, Service. Many of these theories directly compare societies to organisms. At the time I was also very much impressed by the infamous, neo-Malthusian *Limits to Growth* systems dynamics model of the world and the more benign Nicholas Rashevsky's *Looking at History through Mathematics,* but neither seemed to address the problem of qualitative structural change. Immediately after college I read, or should I say *decoded*, Habermas' *Communication and the Evolution of Society*, which in parts really attempts to grapple with this problem.

> In this formulation too it remains unclear what mechanism could help to explain [social] evolutionary innovations. The postulated learning mechanism explains the growth of a cognitive potential and perhaps also its conversion into technologies and strategies that heighten productivity. It can explain the emergence of system problems that, when the structural dissimilarities between forces and relations of production become too great, threaten the continued existence of the mode of production. *But this learning mechanism does not explain how the problems arise can be solved.* (Habermas, 1979, pp.145-146, italics added)

Ultimately, I believe, *all* evolutionary explanations must reach this resting point, that the theory can specify mechanisms for the solving of problems, but the actual solving of the specific problems cannot be incorporated in the theory itself. This requires activity in the real world, an activity outside the model. If this view is correct, there can be no exhaustive theory of function, no enumeration of all possible functions.

Historically, conceptions of biological organization and social theory have mutually affected each other, in Aristotle and Plato, in the effect of Adam Smith's economics on Darwin's evolutionary ideas (Greene, 1981), in the current interest in mutualist, "evolution of cooperation" models (Taylor, 1976; Axelrod, 1981, 1984), and in connections between cybernetics and social self-organization (McEwan, 1963). A long line of social theorizing has emphasized "mutual aid" as a social organizing principle, and sought alternatives to centralized hierarchies of power, instead advocating decentralized, distributed social control. In such theories social coordinations emerge from the free action and association of self-governing agents, who are autonomous beings capable of learning from their mistakes. Social coordinations thus constructed enable individuals to achieve goals which otherwise would be unattainable; a greater realm of freedom is thereby achieved by the formation of higher-level voluntary cooperation. In this evolutionary, "order-from-chaos" worldview, a thoroughly democratic social order can only arise from the bottom upwards, as a true emergent of the desires of individuals and

communities. Social self-organization, rather than hierarchical, top-down specification of behavior, becomes the basis for an anarchistically constructed, voluntarily-stabilized social order, one in which further evolution and self-organization are facilitated for the sole purpose of enhancing the lives of its citizens. While this dissertation seems quite abstract and far removed from the nitty-gritty, day to day struggles that expand the realm of freedom, I hope that its ideas in some small indirect way ultimately add to that purpose.

Chapter 1 Introduction

> The would-be model maker is now in the extremely common situation of facing some incompletely defined 'system,' that he proposes to study through a study of 'its variables'.' Then comes the problem: of the infinity of variables available in this universe, which subset shall he take? What *methods* can he use for selecting the correct subset? (W. Ross Ashby, 1965, p. 343)

> Any machine is prisoner of its input and output domains. (Newell, 1980, p.148)

> There is little use in devising a system of thought about the nature of the trap if the only thing to do in order to get out of the trap is to know the trap and find the exit. Everything else is utterly useless: Singing hymns about the suffering in the trap ... or making poems about the freedom *outside* of the trap, dreamed of *within* the trap ... *The first thing to do is to find the exit out of the trap*. The nature of the trap has no interest whatsoever beyond this one crucial point: WHERE IS THE EXIT OUT OF THE TRAP? (Wilhelm Reich)

Where is the exit out of the trap?

Bound by our notational schemes and mathematical imperatives, our models lie trapped within the bounds of our observables. Seen from within the trap, there are no other alternatives, no other symbol primitives, no other syntactic alternatives, no other observables. *In the world of our models, nothing else is possible*. The world is closed, limited to what we have already defined and specified. To enlarge this world, to briefly escape the confines of this trap into yet another, larger one, we must go outside our present trap: to formulate new notational schemes, to specify new symbolic operations, to find new linkages with the external world.

A predictive model consists of a set of observables, their associated means of measuring the world, and a set of mathematical relations on those observables. Much discussion from mathematics, philosophy of science, and machine learning has dealt with the problem of automating the search for the appropriate relations between observables necessary to solve a given problem. We by no means have the sorts of efficient search strategies we would like, but there are a number of effective methods we can use to arrive at adequate if not optimal solutions. Once we have a notation, we can do a search. But how do we find a notation? How do we construct a new notation?

Most generally, we find new observables, new linkages to the world, new primitive definitions. The question of how we go about finding the observables for our model has been virtually ignored because it lies outside the domain of well-defined, mathematical relationships. This problem, however, is the most important of the two because the observables, the measurements, define the relation of the symbols in the model to the world. Measurements give signs meaning by linking them to the world in a stable way. We are always trapped within the sets of observables we are using at a given time.

How do we break out of this trap? Populations of biological organisms do it by evolving new sensors, allowing them to make more distinctions on the world, thereby expanding their set of available observables. The immune system functions as an evolutionary system in a single organism, by adaptively constructing arrays of molecular sensors through variation and selection. Similarly, throughout history scientists have constructed new measuring instruments which allowed them to apprehend greater and greater portions of the world, to make more distinctions on it.

The salient difference between computers and robots on one hand and biological organisms and human beings on the other is that the latter have the capacity to construct new observables, thereby constructing their relationship to the world around them, while the former are prisoners of their fixed, pre-specified observable domain.

A theory of the role of symbols in both biological systems and in computational devices is necessary if we are to see these distinctions clearly. And we must see these distinctions *very* clearly if we are to design devices which adaptively define for themselves and ultimately for us what their symbols *mean*.

And as we shall see, what they mean is not only how they are connected to the world through observables, but also how they are connected through actions. If the problem of new observables is yet barely recognized, the problem of new controls, new actions, still lies completely unexplored.

If such devices are to solve problems that even we cannot yet adequately characterize, then our device must be able search for possible solutions outside the realm of what we already know and can specify. Our device must be capable of some degree of autonomous action relative to our specifications, it must have *emergent properties* relative to our model, our expectations, of its behavior.

Such devices would be useful to us as a last resort in ill-defined problem situations, where we lack even a plausible rudimentary definition of the problem at hand. This dissertation is about recognizing various levels of problem definition and the sorts of devices which are appropriate for each type of problem situation. Problem-solving in ill-defined situations involves the selection of observables and primitive actions necessary to adequately characterize the problem at hand such that it can be solved. At a very basic level, it involves how the problem will be *encoded* and what kinds of actions will be needed to effect a solution.

Why build devices with emergent semantics?

One might well ask why we would find devices with unpredictably open-ended behavior desirable. After all, this is the antithesis of the completely controllable, completely predictable machine. One answer is that we would not, in fact, want to create *completely* autonomous devices, but ones for which we can control the degree of constraint, such that we can loosen our specificational constraints to let them explore new realms or tighten them when we need predictability, reliability, speed, and efficiency. There are reasons to believe that such devices would help us get along in *our* world, to expand our repertoire of percepts and actions, and at the same time help us to understand the structure of that world. While engineering, scientific, and philosophical goals, being intimately bound up with each other, cannot be fully separated, we can sketch roughly the usefulness of emergent devices for each domain.

The practical reasons for engineering open-ended, emergent learning devices are manifold, although it may take several decades before these technologies come into their own. One can see potential applications in adaptive prosthetic devices. These would include artificial adaptive eyes for the blind, artificial adaptive ears for the deaf, and the adaptive addition of other sensory modalities. These sensors would be adaptively tailored for the particular needs of the users. Evolutionary robotics, building robots which selected their features and actions contingent upon their performance, might be a means of solving the frame problem in artificial intelligence, thereby allowing robots to behave more like autonomous biological organisms and less like competely specified machines. Deep-space scientific exploration, which requires considerable foresight in anticipating the kinds of measurements one would need to make many years in the future, would be an ideal application. Instead of sending up fixed sensors, an adaptive construction system for a large range of sensors could be incorporated into space probes. Functional flexibility would be achieved with a fraction of the materials needed to send up the whole array of sensors. Only those sensors which proved useful at a given point in the mission would be constructed, and such sensors could be dismantled and used for raw material to build other ones once their utility had been exhausted. One approach would be to design artificial immune systems, utilizing either proteins or other types of polymers, which could recognize a large number of macromolecular substrates by adaptively producing an array of molecular antibody probes. We would in effect be sending up an automated molecular biology lab capable of utilizing monoclonal antibody techniques. Antibodies produced by such artificial immune systems might be conjugated to drugs form the basis for drug delivery or they could be used as enzymes in their own right with the attendant medical and bio-technical benefits. Automatic pharmaceutical search might be realized using these means (see chapter 10 for other possibilities).

Sensors and effectors would be constructed for task-specific domains, adaptively tailored to a particular problem context. Open-ended learning devices would perhaps eventually perform tasks more flexibly, improvising interpretations when explicit instructions are lacking. An open-ended learning process would mimic in many respects the process of scientific and technical innovation. Autonomous, adaptive construction of new sensors would parallel human construction of new measuring devices. Adaptive construction of new effectors would correspond to the construction of new prototypes in technological innovation, making new actions possible. We could directly utilize the sensors and effectors created by such a process in scientific and technological domains.

Devices with emergent semantics might be necessary for artificial intelligence to break out of its present logic-driven, straightjacket. Symbolically-based artificial intelligence is at an impasse because of the inherent, static nature of completely encoded frames ("the frame problem" or "the robot's dilemma" as to what to do next, Pylyshyn, 1987). Consequently programmers must completely characterize their problem domain and specify the terms of their solution. Machine learning techniques can reduce some of the syntactic specification needed, but the semantic problem of the definition of the domain still remains. Gordon Pask in the late 1950's recognized that feature specification is a fundamental problem for all learning devices:

> The least requirement of a learning device is that it retains a model of its surroundings gleaned by sensory inputs which register the events which it is specified as able to recognize. Thus, any machine, like any other observer, including ourselves, is limited. Conveniently, its perception is quantized by specifying "sensory events" defined in a "sensory continuum". The model it makes of its surroundings is not isomorphic, that is, in one to one correspondence with real elements. At best the model is mathematically a homomorphic representation which is consistent, though in many to one relation with the real world.
>
> The form of the model, which is determined by the transformation mapping a set of states of the surroundings into a set of events which the machine is able to recognize and register, is decided by its designer. In other words, *the designer* provides a relevance criterion. (Pask, 1958a, p.772, italics added)

AI systems presently can be only as good as their designers' relevance criteria (or "primitive features"). Devices capable of adaptively constructing new sensors and effectors, and consequently new perceptual features and actions, could break through this impasse. For the first time devices would have a measure of autonomy in deciding how to define the problem at hand.

Scientifically, the construction of such devices would prove useful for grounding our theories of cognitive evolution and learning in simple, concrete, potentially understandable physical systems. Some theoretical biologists have looked to simpler organisms as more tractable biological substrates for the analysis of the rudiments of cognitive processes (see chapter 2), and this is a fruitful direction for analysis, but for advancing physical understandability there is no substitute for a device which we have ourselves constructed. The concrete implementation of the processes which generate and stabilize external semantic relations would necessarily consist of the physical means of making distinctions and creating new ones. This implementation, whether successful or not, would clarify many of the issues surrounding the problem of formation of external semantic primitives by grounding the discussion in terms of relatively simple physical devices. Theories of perceptually-based semantics and performance-based pragmatics could be constructed from the concrete situation. Emergent devices would provide a potential conceptual route for breaking out of pre-specified, encoding frames and the purely logical, ungrounded "possible world" constructions of model-theoretic semantics.

Philosophically, this would involve returning to a view of scientific theories in which observables and mathematical relations are once again distinct, to the framework developed by Schlick (1925), to Goodman's *Structure of Appearance* (1951), and the Carnap of the *Aufbau* (1928), before the latter's system turned completely away from observation and toward purely syntactic, platonic modes of explanation (Goodman, 1963; Shanker, 1988, p. 170-171). Discussion of concrete devices, as opposed to discussion of English sentences or logical propositions, should aid in the clarification of many of the

concepts bound up with cognitive processes. Perhaps most importantly, construction of physical devices based upon philosophical premises affords a pragmatic testing of those philosophical systems. Rationalism has had its chance, having been embodied in symbolic artificial intelligence, and has been found wanting:

> 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 commonsense world and of representing that theory in a symbol manipulator had run into just the difficulties Heidegger and Wittgenstein had discovered. (Dreyfus & Dreyfus, 1988, p.34)

Perhaps now is the time to try out the alternatives.

A summary of the argument

This dissertation explores the prospects for designing devices to cope with ill-defined problems, problems where the observables and controls needed to solve the problem are unknown. On a very general level, this involves the relation of symbols in minds, in organisms, in artificial devices to the world at large. The field of semiotics has attempted to grapple with the bewildering array of symbol systems in the world: different types of symbol vehicles, differing purposes and functioning, even comparative analysis of symbol systems across species. What we are after here is an analysis of how symbols come to acquire real world functions, so that these relations, mechanisms, and concepts can be brought to bear on the problem of constructing devices which generate new functions.

The first chapter introduces some of the general features of symbol systems. Symbols necessarily involve discrete, distinguishable sets of alternatives implemented through utilize time-independent structures. The form of the symbol is arbitrarily related to its consequences, and the definite relationship that symbols come to have with their contexts is a historical process of successive selection and constraint. According to the pragmatist theory of communication developed by Charles Morris, symbols can be analyzed in terms of three irreducible, orthogonal types of relations: syntax, semantics, and pragmatics. Syntax is the relation of symbols to other symbols, semantics, the relation of symbols to their real world contexts, and pragmatics, the usefulness or purpose of a given symbol, its relationship to its user. Figure 1.1 illustrates these relations for DNA.

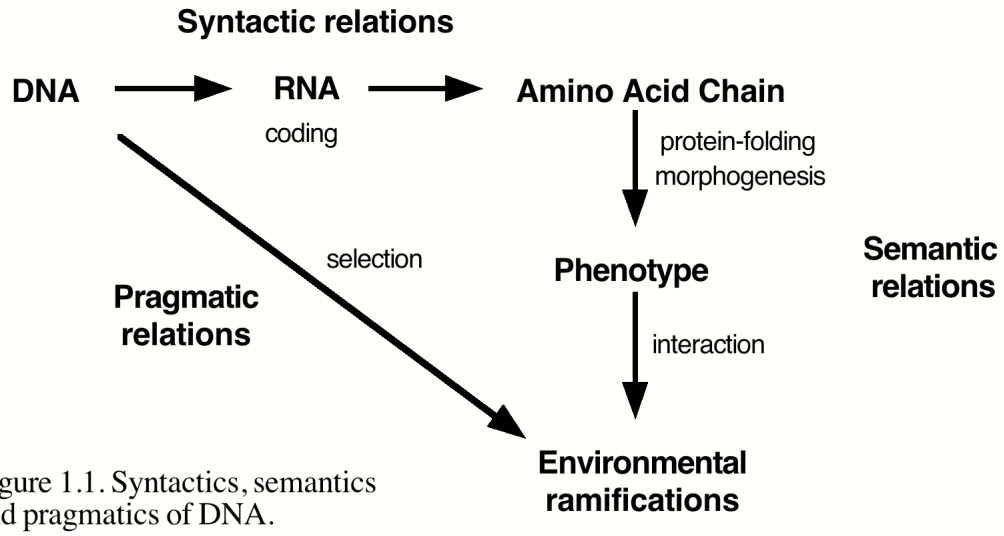

Figure 1.1. Syntactics, semantics and pragmatics of DNA.

The general alternative to this schema is the concept of symbol as reference, which involves a correspondence theory of truth, a realist ontology, and a number of neo-platonic assumptions regarding the relation between logic and order in the world. This worldview has some validity in completely

encoded, unambiguous, logic-based domains of pure mathematics, but it is not very useful for analyzing symbols in natural systems, where the correspondences are incompletely known. For biological systems the pragmatist notion of symbols as instructions is much more appropriate. Thus conceived we can see how symbols enable communication, coordinated activity, and even a public realm of experience. Science becomes possible through symbolic mediation.

Symbols are ubiquitous in biology, from DNA to hormonal messengers to neural encodings. The very basis of biological organization is wrapped up in symbolic participation in inheritance and morphogenesis. Conversely, virtually all symbols are connected with living things. We cannot understand one without the other.

The second chapter examines some of the general theories of biological organization which deal with the problem of information and meaning in biological systems. Biological information theory, thermodynamic theories, analysis of various symbolic functions, and computationalist theories of biology are assessed and addressed with respect to their analysis of symbolic function. None of these theories adequately incorporates syntactic, semantic, and pragmatic relations. A "cognitive biology," which would address these issues is discussed and developed. In this scheme, syntactic relations play the role of coordinating the mapping between an organism's percepts and its available actions. Semantic relations, the relation of the organism's symbols to its environment, are implemented by the organism's sensory apparatus and its effectors. Pragmatic relations are here involved with "teleological" relations: mutation, the generation of alternative organisms, the selection of those best adapted to their environments, and the propagation of organisms similar to the survivors. When all of these relations are present in properly interconnected mechanisms, we have a *semantic closure* which enables the co-evolution of the symbols and the physical biological processes they constrain (figure 1.2).

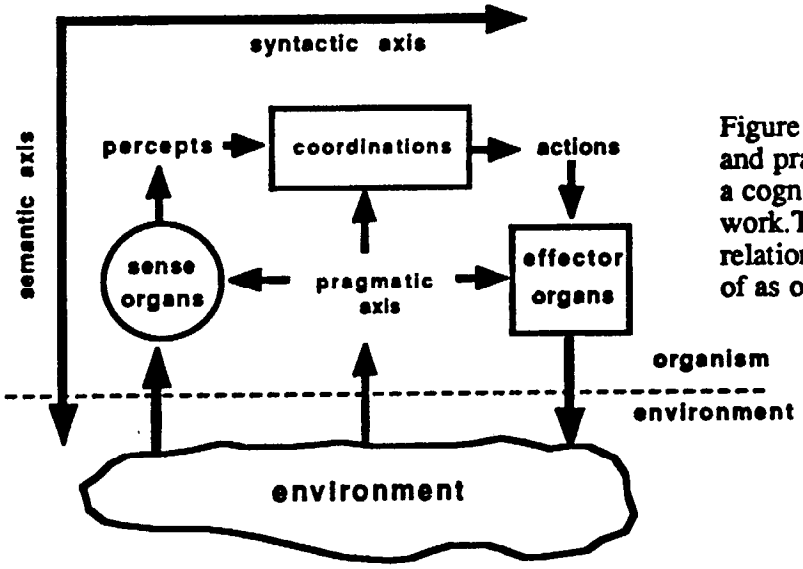

Figure 1.2 Syntactic, semantic and pragmatic relations within a cognitive biological framework. The three semiotic relations should be thought of as orthogonal axes.

The fourth chapter examines the parallels of this set of epistemic relations in physics. The gradual inclusion of symbols into ever more explicit accounts of the modelling relation was carried out by von Helmholz, Hertz, Bohr, and the theoretical biologist Robert Rosen. Once again we see three irreducible axes: the syntactic involves the mathematical computations of the formal part of the model, the semantic involves the construction of measuring devices and the measurement process itself, and the pragmatic involves the free choice of the observer as to what to measure, what to model (figure 1.3). This irreducibility is related to von Neumann's arguments concerning the measurement problem in physics.

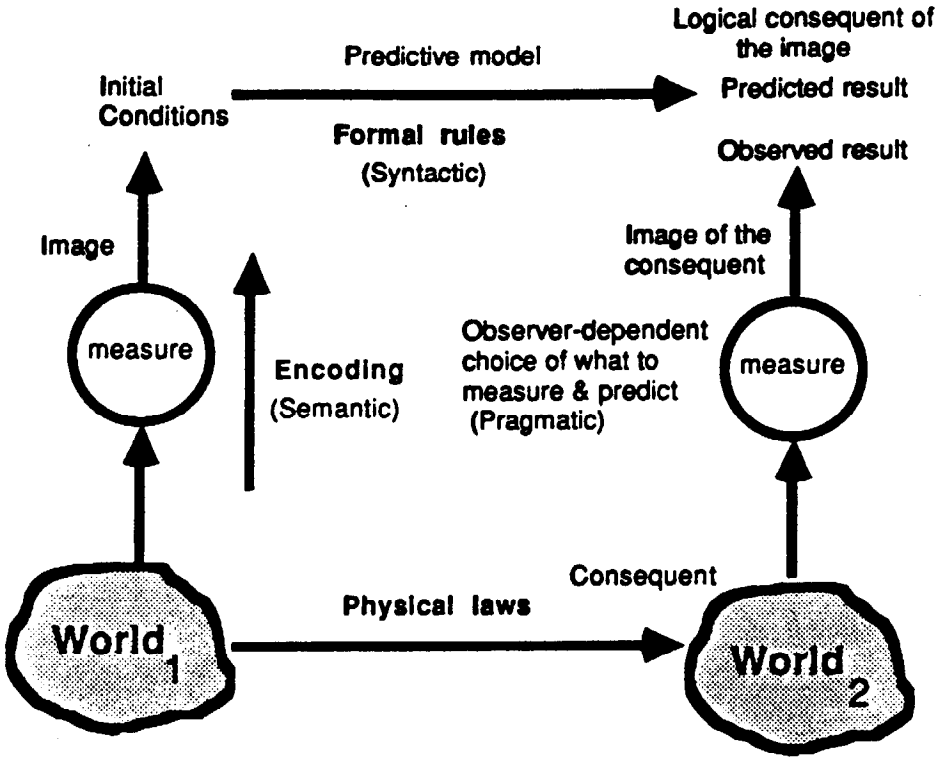

Figure 1.3 The role of symbols in the modelling relation.

Physical models of symbolic processes can also be used to shed light on the nature of their construction. Howard Pattee has argued that symbols necessarily permit multiple descriptions, as the operation of time-independent, non-holonomic constraints or as the operation of time-dependent physical laws. These multiple, formally irreducible descriptions yield a complementarity between descriptions of rate-independent, rule-governed processes and rate-dependent, law-governed ones. This complementarity of descriptions extends to "symbolic" and "nonsymbolic" processes and to the physical description of computations and measurements. Again, we come to a situation involving semantic closure, this time framed in terms of the coupling of nonsymbolic, rate-dependent processes with rate-independent symbols.

We want to be able to use these distinctions to analyze the functional properties of various types of devices. To do this we need to recognize the basic functionalities we will use in our devices. In the fifth chapter the symbolic-nonsymbolic distinction is used to demarcate four irreducible primitive symbolic-nonsymbolic transformations: computation, measurement, control, and nonsymbolic interaction (figure 1.4).

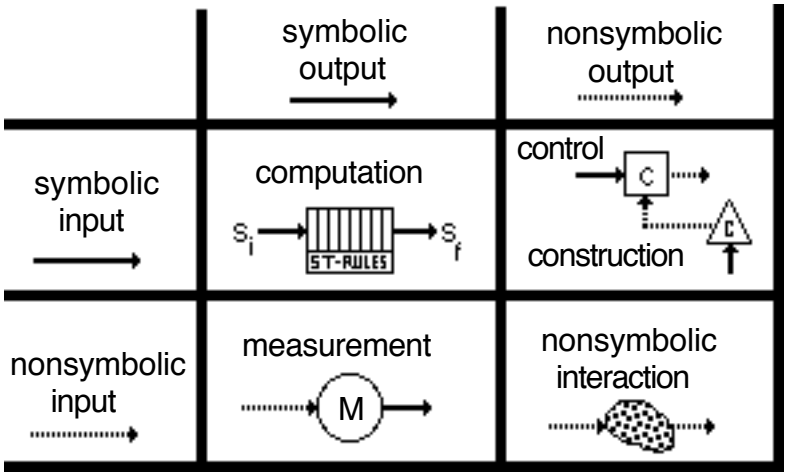

Figure 1.4 Transformations between the symbolic and the nonsymbolic.

Decomposing device-environment (or organism-environment) relations into these primitive operations allows for the functional analysis of various types of devices. The transformation of computation, which involves only the deterministic mapping of symbol strings into other symbol strings, is seen as a completely deterministic, syntactic operation. The transformations of measurement and control, which mediate between the symbolic realm of the device and the nonsymbolic world at large, ultimately will be seen as determining the real world semantics of the symbols used in computations. Some of the correspondences between these transformations and their observed state-transition behavior are outlined. This mapping should make it possible to unambiguously distinguish between the different transformations relative to a given, well-defined observational frame. Thus we can have systems-theoretic criteria for relating our cybernetic mechanisms to their observed behavior.

The basic structure of our devices, the causal dependencies between the functionalities, will determine which parts are plastic and which parts are static. Adaptivity is the changing of internal structure so as to better perform in an external environment. We can structure devices so that they improve their performance with experience by making the plastic functionalities contingent upon performance. The performance dependent functionalities can be either in the syntactic or the semantic realm or both. The sixth chapter develops a taxonomy of adaptivity in which three different types of devices are distinguished: nonadaptive devices, syntactically-adaptive devices, and semantically-adaptive devices. These device types are a consequence of which structures are plastic and which are fixed (figure 1.5). *Formal-computational devices* have only fixed syntactic parts, and no inherent semantics. *Formal-robotic* devices have both fixed syntaxes and semantics. Because of their completely static structures, both types are non-adaptive. *Adaptive devices* have adaptive syntaxes. *Evolutionary devices* have adaptive semantics. *General evolutionary devices* have both adaptive syntaxes and adaptive semantics.

device type	plasticity	syntax	semantics
formal-computational	fixed syntax	performance-independen	No inherent semantics
formal-robotic	fixed syntax fixed semantics	performance-independent	Fixed sensors and effectors
adaptive	adaptive syntax fixed semantics	performance-contingent	Fixed sensors and effectors
general evolutionary	adaptive syntax adaptive semantics	performance-contingent	performance-contingent syntax performance-contingent semantics

Figure 1.5 A taxonomy of types of adaptivity.

Biologically, the three levels roughly correspond to 1) the (relatively) fixed behavior of the biological individual, 2) microevolutionary selection within a fixed set of genes, and 3) macroevolutionary production of new structures and capabilities. In systems theoretic terms, these levels correspond to 1) fixed parameter models, 2) free parameter models, and 3) free observable models. The relationship of this taxonomy to other ones is briefly outlined: the three knowledge levels of Piaget; the state-determined systems, Markov systems, and self-organizing systems of Ashby; the taxonomy of adaptive devices of Gause & Rogers, the survey of adaptive devices conducted by Barto and Sutton, and the Inquiring Systems of C.West Churchman. Chapters 7-10 analyze the capabilities and limitations of each of these device types in detail.

Formal-computational devices are discussed in chapter 7. These devices do not have any input or output transducers, they operate on uninterpreted symbols. They are not considered to have completed a computation until they halt, and they must halt within some specified, relevant time interval. An extended discussion of the definition is given to avoid confusion between these devices, which resemble the computers we use every day, and various neo-platonic ideal devices (e.g. arbitrary Turing machines) with indefinitely extendible tapes and infinite time allotted for processing. Many of the issues involved with the definition of computational devices have been debated within the foundations of mathematics, and the physically-based strict-finitist, constructivist view adopted here is elaborated in appendix 1.

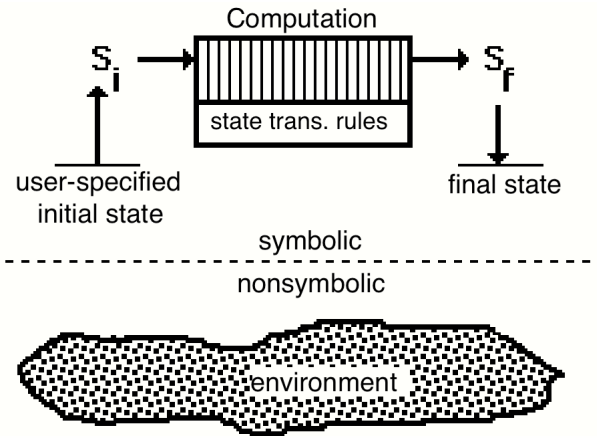

Figure 1.6 Structure-function diagram of formal-computational device. Note that symbols manipulated by the device are not directly connected to the external world save through human intervention and specification.

Formal-computational devices are semantic-less and syntactically static. Their limitations flow from a lack of inherent semantic connections with the world and the static, nonadaptive nature of their syntactic operations (figure 1.6). On the other hand, they will reliably implement the same symbolic input-output mappings, and are therefore very well suited for complex, fixed syntactic operations. Because of their completely symbolic nature, formal-computational devices can only deal with problems which are already
completely encoded into symbols. Because of their completely nonadaptive nature, everything that they do must in one way or another be completely specified. Formal-computational devices cannot create genuinely new syntactic or semantic primitives. The best that they can do is to combine pre-existing primitives into new combinations, as in Lenat's "mathematics discovering" program. Many related critiques of computers and formal systems have been mounted over the years and they are here connected to adaptivity considerations.

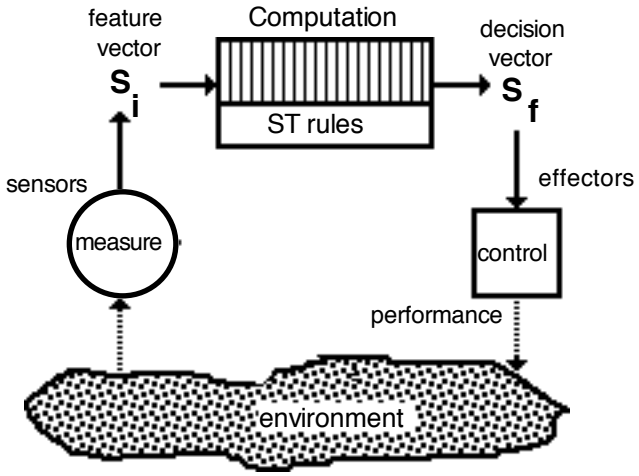

Figure 1.7 Structure-function diagram for a formal-robotic device. The internal structure of the device is unchanged by its interactions with its environment.

The properties of robotic devices are discussed in chapter 8. Robotic devices are similar to formal-computational ones, except that they have fixed sensors and/or effectors which connect them directly to the world, thereby giving them inherent external semantics (figure 1.7). Robotic devices do not have to depend upon human beings to provide the interpretations for their symbols. Having both fixed syntaxes and semantics, robotic devices can solve problems which are not already in symbolic form, such as welding parts together on an assembly line. Still, robotic devices are nonadaptive because they do not possess any structural plasticity. They cannot alter their internal structure to optimize their performance.
Adaptive devices are a different story. Adaptive devices alter their syntactic, computational parts based upon their past experience and performance (figure 1.8). Roughly speaking, neural nets, trainable classifiers, Boltzmann machines, genetic algorithms, and connectionist associators are all adaptive devices when they do have genuine external, real world semantics. Completely simulated devices, strictly speaking, do not make measurements or effect control operations. Embedding these devices within simulations effectively makes them part of a completely syntactic system, unless the observer has limited access to the simulation environment. This thorny issue is discussed at some length because it relates to whether these devices can be effectively implemented via computational simulations, thus collapsing the category into that of formal-computational devices. The conclusion here is that these devices, being dependent upon measurements, performances, and evaluations, cannot be so reduced.

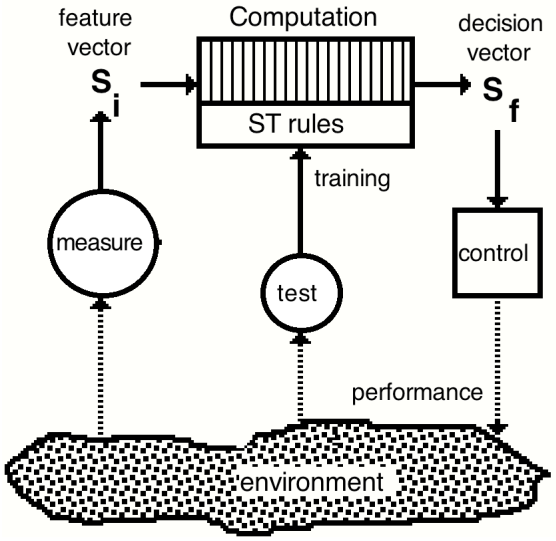

Figure 1.8 Structure-function diagram for an adaptive device. Here the structure of the computational part is dependent upon its performance.

The advantages of adaptive devices lie in their flexibility and ability to adapt to unforeseen situations. To the extent that a given functionality can adapt itself to the exigencies of a specific situation, the designer is freed from having to directly foresee and specify the appropriate behavior. This adaptation, however, is ultimately limited by the features and actions available to the device. The behavior of the device is semantically bounded: it must take place within the confines of what its sensors and effectors can do.

Evolutionary devices, those with adaptive semantics, can enlarge the confines of the semantic realm by adaptively constructing new sensors and effectors (figures 1.9a, b).

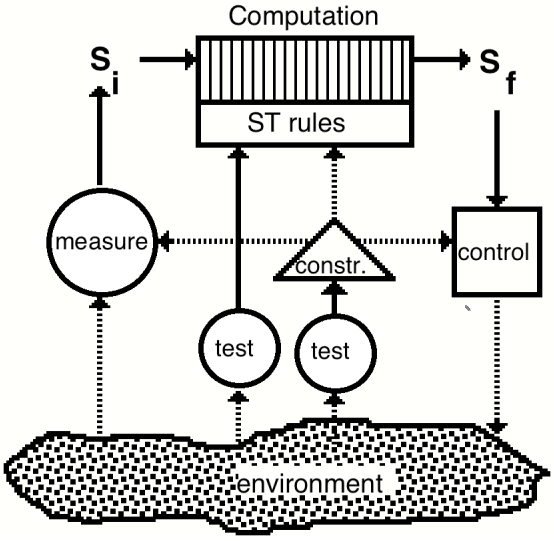

Figure 1.9a. Structure-function diagram for an evolutionary device with adaptive semantics but a nonadaptive syntactic part.The immune system is an example of this kind of device.

New sensors and effectors mean new primitive features and actions. General evolutionary devices have the capability of constructing all of their parts: they are both semantically adaptive and syntactically adaptive. Chapter 10 gives some examples of such devices: the evolution of new sense organs and effectors in biological evolution, the mutation and selection of specific antibodies during the immune response, the construction of tools and sensing instruments by human beings. We radically enlarge our perceptual and behavioral repertoires by constructing artificial means of interacting with the world.

Technology can be seen as the construction of prosthetic devices of various sorts. To my knowledge, only one person, Gordon Pask, has proposed such a device with the explicit goal of automating the observable generation process. In the late 1950's he constructed an electrochemical device apparently

with the intention of physically implementing an analog learning network, which became sensitive to other kinds of perturbations, such that it could be tuned with the appropriate rewards. Apparently, this is the first and only evolutionary device in existence. Were we to build one today, how would we go about it?

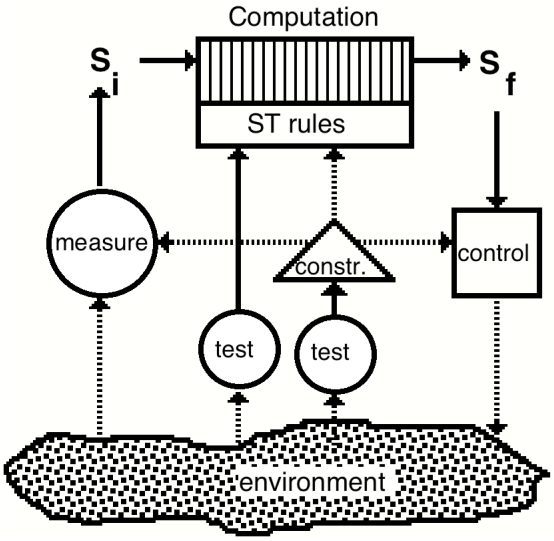

Figure 1.9b. Structure-function diagram for a general evolutionary device. All parts of the device are adaptively constructed, yielding both syntactic adaptivity and semantic adaptivity.

One of the requisite properties of an evolutionary device is that it contain a construction system for physically building the sensors and effectors. In biological organisms, the transcription-translation-protein-folding process serves as the construction system. Some of the properties of this system are that it must have a heritable, symbolic component, it must be able to operate reliably, and the range of potential structures must be sufficiently rich to ensure a reasonable chance for functional improvement. The richness of the space of structures and the sensitivity of these structures to environmental influences will determine the degree to which the evolutionary device is capable of open-ended behavior. Lastly, there is the question of whether we as human beings unaided by tools or language are ourselves evolutionary devices. This is an open question and one which is highly dependent on one's observational frame of reference. Tentatively, I conclude yes.

In chapter 11 the capacities and limitations of the various device types are reviewed and summarized (figure 1.10). Various strategies for problem-solving and design can be linked to the different device types. If a problem can be effectively encoded and its context is well understood, then writing a fixed computer program is a viable problem-solving option. If the problem is of a nonsymbolic kind (e.g. machining metal), but the context is fixed and well understood, construction of a robot is warranted. If one has the appropriate feature and action primitives needed to effect a solution, but the context is not known well enough to optimize their coordination beforehand, then an adaptive device is a reasonable approach. In situations where the basic features and actions needed to effect a problem solution is not known, and/or the context is poorly understood, then an evolutionary device might be a possibility. As the problem-solving process proceeds from less constraint to greater constraint, there will generally be a progression from evolutionary to adaptive to robotic and computational devices as semantic then syntactic relations are successively frozen in their optimal forms.

device type	plasticity	capacities	limitations
formal-computational	fixed syntax	reliable execution of pre-specified rules	limited to pre-specified rules and states
formal-robotic	fixed syntax fixed semantics	reliable execution of fixed percept-action combinations	no feedback or learning from environment
adaptive	adaptive syntax fixed semantics	performance-dependent optimization of percept-action coordination	limited to percept & action categories fixed by the sensors & effectors
general evolutionary	adaptive syntax adaptive semantics	creation of new percept & action categories; performance-dependent optimization within these categories	time to construct & test new sensors & effectors may be very long

Figure 1.10 Summary of capacities and limitations of the device types.

Up until now, we have not dealt explicitly with the problem of emergence. If devices are to useful in expanding our own faculties, they must exhibit emergence relative to our expectations, they must surprise us. They must transcend the notations we have developed to describe them. Chapter 12 deals with the problem of emergence and the relation of the various devices to different types of emergence. On the contemporary scene there are three major conceptions of emergence, *computational emergence*, *thermodynamic emergence,* and *emergence-relative-to-a-model* .

Theory	Account of the origins of order	Ontology	Research program
Computational Emergence	**Order-from-Order** Macro-order from micro-determinism Macro-indeterminism through mathematical chaos	**Discrete Universe Monism** Microscopic Deterministic Rules	**Realize emergent behaviors through cellular automata and evolutionary computer simulations (Langton, Toffoli)**
Thermodynamic Emergence	**Order-from-Noise** Discrete macro-structures (symbols) from continuous micro-processes New structures emerge through fluctuations	**Continuous Universe Monism** Continuous Physical Laws	**Develop a thermodynamic theory to describe how emergent structures can arise far from equilibrium. Apply this to biological & social systems & upwards (Prigogine, Iberall)**
Emergence relative to a model	**Form-from-formlessness (Order-from-Chaos)** Processes of linking symbols to the world result in new functions and extend the realm of symbolic activity	**Symbol-Matter Dualism** Dualism between Definite (measured & Indefinite (un-measured)Entities	**Realize emergent functions through the construction of semantically adaptive devices, augment the capabilities of the observer (Pattee, Rosen, Pask)**

Figure 1.11. Contemporary conceptions of emergence

Computational emergence sees emergence as an "objective" process in which deterministic micro-processes form higher order, macroscopic emergents, much in the same way that surprisingly complicated patterns can arise during a cellular automata simulation. Much of the artificial life

community seems to adhere to this conception. There is a propensity in this framework to see the emergence *as being realized in the cellular automata simulation itself*, not as some might think in the mind of the observer or in the observer-machine relationship. The greatest difficulty with this conception is that it does not specify beforehand what the expectations of the observer might be--what would or would not be emergent behavior. If mere surprise is the sole operant criterion, then virtually *all* computations are emergent, until, perhaps, one has seen the same one enough times. Here the ubiquity of emergence is proportional to the vagueness of the observer's expectations.

Thermodynamic emergence encompasses most of the large body of work in nonequilibrium thermodynamics, dynamical theory, and information theory. The paradigm case is the orign of life. Concepts of information, entropy, energy dissipation, structural complexity, and catalytic networks are employed to construct a theory regarding the origins of new complex structures far from equilibrium.

The emergence-relative-to-a-model view developed by Robert Rosen is to see emergence as behavior relative to a formal predictive model of what is expected to happen. Here the model and the observables must be well defined, and the behavior of the device must be completely captured by the model before the emergent event(s). The emergent event is a deviation of the behavior of the physical system under observation from its predicted behavior. In contrast to the ill-defined character of computational emergence, emergence-relative-to-a-model has the advantage of being precise, well-defined, and unambiguous in its application. It also has the advantage of being a *functional* theory of emergence by giving an account of how new basic functions of the observer--measurements, computations, and controls--can come into being.

Property	Device type	Model change needed
Nonemergence	**Formal-computational** **Formal-robotic**	Device reliably conforms to model's expectations; (no change needed)
Syntactically emergent	**Adaptive**	Device diverges from model; parameter adjustments alone are inadequate; new observables are needed
Semantically emergent	**Evolutionary** **General evolutionary**	Device diverges from model; parameter adjustments alone are inadequate; new observables are needed

Figure 1.12. Relations between types of emergence and types of adaptive devices.

The various device types of the adaptivity taxonomy can be related to this conception of emergence. At the start, our formal model of the device will have the same observed state-transition behavior as the device itself and our measuring devices for the model will have been calibrated to agree with the sensors of the device. If we allow the device to run, interacting with the environment, three things can happen (figure 1.13). First, our model can continue to completely predict the behavior of the device, in which case the device is a formal-computational or robotic device. Second, the behavior of the device can deviate from our model, but an adjustment of model parameters is sufficient to completely recover the behavior. In this case only a syntactic adjustment was necessary for our model, so our device appears to us as an adaptive one. Third, the behavior of the device can deviate from our model, but it is necessary to change the measurements we are making in order to recover the behavior. In this case a semantic adjustment was necessary, and we appear to have an evolutionary device on our hands. If the behavior

cannot be recaptured (or if it cannot be fully captured in the first place), then we have a stochastic system of some sort, and there is very little definite that we can say about its syntactic and semantic structure.

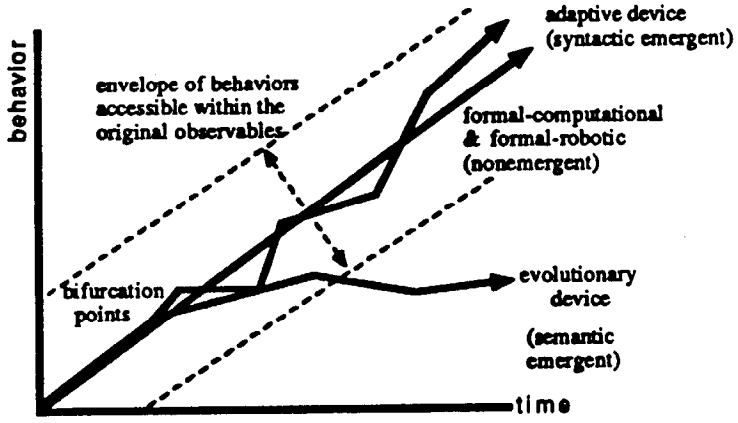

Figure 1.13 Divergence of the behaviors of adaptive devices from fixed models of them.

The new semantic relations created by an evolutionary device can amplify the capacities of a human observer by a process of linking the human's observables to the new observables of the device. This process, described by Rosen, is related to how we are able to appropriate objects external to ourselves to effectively integrate them into our bodies as extensions of our senses and our muscles. In effect we form a syntactic linkage with an external measuring device, thereby extending outwards the realm of syntactic operations to the measuring device, which has become our new semantic boundary with the world.

Finally, we can speculate on why semantic processes appear to us to be open-ended and incompletely defined while syntactic ones appear to us to be definite and closed. This is related to the apparently necessary character of syntactic relations versus the apparently contingent nature of semantic ones (figure 1.14).

Definite-symbolic **Explicate**	**syntactic**	**computations**	Exact comparisons between all elements possible; Given definite nature of elements and finite numbers, set of possible symbol manipulations is closed
Partially-definite	**semantic** **pragmatic**	**measurements** **controls** **constructions** **selections**	Exact comparisons between measurements & controls only possible through symbolic part; Partially definite elements yield indefinite, open set of distinguishable possibilities
Indefinite **Implicate**	**nonsymbolic**	**nonsymbolic interactions**	No exact comparisons are possible, so set elements cannot be distinguished at all; Completely indefinite elements yield no set, since no distinctions can be made

Figure 1.14. Definiteness, indefiniteness, and closure.

The relative closure of the repertoires of the various device types can also be analyzed. What can be known about a device's behavior given knowledge of its structure, its adaptivity type? It can be argued that, for finite sets of symbol primitives, all of the input-output functions between those sets can be enumerated. For a finite symbolic domain, we could say that the set of all possible computations is closed. This is manifestly not the case, however, for measurements and controls because these

operations lie outside the well-defined domain of symbol manipulations. While a computation, being a completely symbolic function, can be completely described by its observed behavior, a measurement, being a partially nonsymbolic function, cannot be so described. Sets of possible measurements are thus indefinite, open, incompletely described, while the set of all computations for a given device is definite and closed. As a result those devices which are semantically adaptive also have a relatively open-ended range of possible functions relative to their nonadaptive counterparts.

The last chapter briefly recapitulates the major ideas traversed and summarizes the major implications of the dissertation.

The appendices are extended discussions of particular points made in the text. The first appendix examines the problem of incorporating potential infinities into any theory of the capacities and limitations of constructible, physical devices. It also examines the role which potential infinities play in theories of language and cognitive science. This discussion is extended into appendix 2, where the concept of "internal semantics" is critiqued. The programs-have-semantics view depends upon making an essential distinction between interpreted and uninterpreted symbols in computers. This distinction, which underlies the level separations of both Newell and Pylyshyn, rests on the difference between a Turing machine (having a potentially infinite tape) and finite state automata (having limited states). But indefinitely extendible tapes do not exist, at least in the physical world, and if only Turing machines with finite tapes are considered, the distinction becomes purely one of notation. Appendix 3 considers the question of whether evolutionary simulations might be able to generate *de novo* primitives, whether they might have emergent properties in the sense used here. Robert Rosen's (1973) analysis of evolutionary models, of transformation and emergent model types, argues that these model types, which cover virtually all evolutionary models in existence, cannot generate new primitives. Various break-out strategies, such as morphogenetic modelling, complexifying the environment, meta-rules, rules to recognize patterns (as in Lenat), mathematical chaos generators, stochastic processes, cellular automata and connectionist networks are examined. Unfortunately, none of them changes the completely closed, syntactic nature of the simulation. Appendix 4 discusses the different use of the terms "prediction," "linearity," and "understandability" in physics and in computer science and the serious confusions which can be caused by their multiple meanings. Appendix 5 examines the perspective of John von Neumann, whose name many advocates of strong artificial life invoke in support of their position. From a close reading of his relevant texts, it is argued that von Neumann's views are diametrically opposed to the reduction of all biological processes to digital-computational ones. At many points, von Neumann points out the limitations and dangers of "the axiomatic approach," and emphasizes the importance of mixed digital-analog processes in biological organisms and in the brain.

Chapter 2 The natural philosophy of symbols

> The logic of things, i.e. of the material concepts and relations on which the structure of science rests, cannot be separated by the logic of signs. For the sign is no mere accidental cloak of the idea, but its necessary and essential organ. It serves not merely to communicate a complete and given thought content, but is an instrument, by means of which this content develops and fully defines itself. The conceptual definition of a content goes hand in hand with its stabilization in some characteristic sign. Consequently, all truly strict and exact thought is sustained by the *symbolics* and *semiotics* on which it is based....
> This mutual relation is not limited to science but runs through all the other fundamental forms of cultural activity. (Cassirer, 1955, pp.85-86)

The ubiquity of symbols

Despite the nearly complete dependence of our entire civilization on the strings of symbols that implement human language as well as those which form the grist for our information technologies, there is still lacking a coherent, comprehensive scientific account of what the symbols themselves are and how they work.

If we are to design devices capable of solving ill-defined problems and clearly communicating their solutions, the devices themselves must have the means of framing the problem, of encoding the problem into symbolic form. To tackle these problems we must have a clear idea of what symbols are, how we distinguish between symbolic and nonsymbolic processes. We must also undertake to understand the origins of symbols, how symbol systems evolved in biological organisms, how biological symbols facilitate biological complexification, how languages evolve in human communities, and how formal languages come to be constructed for specialized problem domains.

Symbols and signs

The related notions of *symbol* and *sign* are epistemological primitives with a long lineage, stretching back to the ancient Greeks (Eco, 1984, p.26). From the start conceptions of symbols have been intimately connected with philosophical debates regarding the nature of the world.

> The term 'semiotic' goes back to the Greek medical tradition which considered semiotic, embracing diagnosis and prognosis by signs, as one of the three divisions of medicine. The Stoics gave semiotic the dignity of a basic division of philosophy co-ordinate with physics and ethics, and included within it logic and the theory of knowledge. The whole of Hellenistic philosophy centered around the semiotic, and in particular the problem of empiricism versus metaphysics was formulated as a problem of the limits of of signifying by signs, the Stoics arguing that there were signs ("indicative signs") which could give necessary knowledge about things beyond the limits of observation; the Epicureans holding that while signs gained their signification through experience, some signs (such as 'atom' and 'void') could, though only with probability, refer to what was not capable of direct observation; the Sceptics questioned the whole edifice of metaphysics on the ground that signs could refer only to that which was observable, serving to recall (as "commemorative signs") that which had been observed even though it was not at the moment of reference directly observable. (Morris, 1946, pp. 285-286)

Modern semiotics attempts to explicate the notions of language, symbol, sign, code, diagram, metaphor, image, icon, and stylization as well as their underlying philosophical assumptions. Each notion has specific consequences for a general theory of sign production and interpretation (Eco, 1976, 1984; Goodman, 1976). The disentanglement of the multifarious uses of these notions is a formidable task, full of hermeneutic pitfalls and linguistic morasses.

Roughly speaking, in conventional usage "symbols" embody meaning, while "signs" have only definite form. Symbols involve semantic functions, while signs are purely syntactic ones. When one speaks of a "sign," one is talking about the form or appearance of the token shorn of any meaning or relationship to the world. The sign can be manipulated as a meaningless object. A symbol, on the other hand, is a sign endowed with meaning by virtue of its history or relation to a linguistic convention. A

symbol is used for its meaning and according to its meaning. In many scientific contexts, however, it is common to talk of "mathematical symbols" rather than "mathematical signs," when the one is talking about the specific tokens involved and not the meaning. Here we will use the word "symbol" to denote the symbol vehicle without implying whether it has meaning attached to it, and the word "sign" to denote the symbol vehicle without any associated meaning or interpretation.

The salient properties of symbols are that they are discrete and distinguishable, they operate independently of time, they bear an arbitrary relationship to the world at large, and consequently, that this relationship can only be accounted for through the history of its formation. More of the "design features of language" could, of course, be added: the properties of duality (multiple levels), productivity (richness of alternatives), displacement (reference to far removed objects and events), reciprocity ("any organism equipped for the transmission of messages in the system is also equipped to receive messages in the same system" (Hockett, 1958)), complete feedback (a speaker can hear his/her own productions), specialization, cultural transmission, learnability, reflexivity (capability for self-description), and prevarication (Lyons, 1977; pp.70-85). We will briefly address only the basic characteristics here. In the next chapter these properties will be related to biological symbols.

Discrete alternatives and distinguishability. Symbols are only possible as discrete alternatives implemented through distinguishable signs via material tokens. Without alternatives, nothing can be communicated, without discreteness, the alternatives cannot be distinguished:

> In other words, phonology builds up a system of oppositions in order to explain the functioning of a number of phonetic presences which, if they do not exist prior to the system, nonetheless are associated with its ghost. Without people uttering sounds, phonology could not exist, but without the system postulated by phonology, people could not distinguish between sounds. *Types* are recognized through their realizations into concrete *tokens*. One cannot speak of a form (of the expression or of the content) without presupposing a matter and linking it immediately (neither before nor after) to a substance. (Eco, 1984, p.23)

The functional category of type is connected to a distinguishable, concrete, material structure which is the token.

Independence of time. Symbols have a time-less quality. Their existence stands outside of time, as the text of an encyclopedia. The symbol strings persist intact, although their interpretation, their relation to the world around them may change. Symbols enable memory.

Arbitrary conventions. Symbols are arbitrary conventions. They are unlike the things to which they are semantically connected. There is no inherent meaning for particular words, they only come to mean anything through their repeated use. They are social conventions.

> An instructive example is the comprehension of our native language. This knowledge is not inborn; for doubtless we have acquired our mother tongue by learning, that is, usage through frequently repeated experience. Children of our nation who were born outside the boundaries of our fatherland, and who grew up among people speaking a foreign language, would have learned another language and have become just as adept in it as we in ours. (von Helmholtz, 1894, p. 249)

History and origins. That symbols are arbitrary, conventional constructions raises the question of how they come to have specific meanings and uses at all. These meanings and uses, unconnected to the symbol by means of necessary law-like relations, must be contingent relations selected over time for their usefulness. The way a child connects particular words with their meanings lies in successive constraint through repeated experience:

> The example of language is instructive also in another respect, for it affords an explanation for the problem of how such a certain and conventional understanding of a system of signs may be obtained, considering that these signs can have only a quite arbitrarily selected effect upon the individual observer, even though the comparative philologist knows how to recognize traces of the connection of individual roots in it. The mother tongue is learned only by using

words. The child hears the usual name of an object pronounced again and again when it is shown or given to him, and constantly hears the same change in the visible environment described with the same word. Thus the word becomes attached to the thing in his memory more firmly, the more frequently both are repeated. (von Helmholtz, 1894, p.249)

Note that this associationist perspective regarding the contingency of symbols and their meanings does not necessarily conflict with the existence of internal psychological structures, be they representations, computations, faculties, schemata, feedback control structures, complex dynamical configurations or even psychoanalytic complexes. It does imply that the brain is capable of recognizing signs in the environment and manipulating symbols internally, at least to some extent. Currently there is great debate in cognitive science over the functioning and origins of psychological structures, and no agreement is in sight. However, these structures, whatever their form and nature, constrain and channel sensory experience, thereby co-determining, along with that sensory experience what structural connections and associations will be made. Whether these structures are "innate" or "acquired," i.e. whether they are "learned" or "evolved" is largely irrelevant to the present discussion, being as it is about the properties of general types of mechanisms, and not about which mechanisms are found in which parts of the brain. From the larger perspective of evolution, *all* structures and functions are "acquired," and it is in this process of the acquisition of new functions that we are primarily interested. Evolutionarily, the mind is no different from any other function:

The point of view of the evolutionist forces us to view mind in the context of other aspects of evolution, to draw parallels with other more mundane forms of adaptation, such as the organs of locomotion and of digestion. In the context of evolution, the mind of the adult human, the object of so many centuries of philosophical studies, ceases to be a mysterious phenomenon, a thing unto itself. Rather, mind is seen to be an adaptive response to selective pressures, just as nearly everything else in the living world." (Delbrück, 1986, pp. 269-270)

The structures of perception, cognition, and action are all adapted to the outer world through either evolutionary refinement or ontogenetic learning. On one of these levels William James' "*Every perception is an acquired perception*" (William James,1892, p.180) *must* hold true, or as Herbert Spencer observed, "What is *a priori* for the individual is *a posteriori* for the species" (in Munévar, 1981, p.78).

Syntactics, semantics, and pragmatics

In my argument here I will use, following Morris (1946, 1955), three irreducible types of relations involving symbols: the syntactic, the semantic, and the pragmatic:

In the study of language [Charles Morris] saw three main levels: *syntax, semantics,* and *pragmatics*. The syntactic properties of an expression are determined only by its relations to other expressions, considered independently of meaning or interpretation....Semantic properties concern the relation of the expression to the world....Finally, pragmatics concerns the relation of the language to the users of that language**;** (van Fraassen, 1980, p.89,)

...*pragmatics* is that portion of the semiotic which deals with the origin, uses, and effects of signs within the behavior in which they occur; *semantics* deals with the signification of signs in all modes of signifying; *syntactics* deals with combinations of signs without regard to their specific significations or their relation to the behavior in which they occur.

When so conceived, pragmatics, semantics, and syntactics, are all interpretable within a behaviorally oriented semiotic, syntactics studying the ways in which signs are combined, semantics studying the signification of signs, and so the interpretant behavior without which there is no signification, pragmatics studying the origin, uses, and effects of signs within the total behavior of the interpretants of signs. The difference does not lie in the presence or absence of behavior but in the sector of behavior under consideration. The full account of signs will involve all three considerations. (Morris, 1946, p. 219)

Syntax, semantics, and pragmatics are perhaps the minimum properties of a symbol-using subject: ability to manipulate symbols (syntax), ability to connect symbols to the world (semantics), and the ability to use symbols purposefully (pragmatics) to attain some end.

Syntactics. Syntax is the set of rules which govern the manipulation of signs. It is the necessary relations of signs to other signs.

A syntactic operation is one in which only the property of type plays a role, and because only type properties are involved it is always possible to specify a rule for a syntactic operation. Syntax is rule-governed behavior describing necessary relationships. Syntactics implement necessary, logical consequences. All uninterpreted formal procedures and computer programs are completely syntactic operations.

Semantics. Semantics is the relation of symbols to the world at large. This relation is not a necessary one, but contingent upon the structure of the interpretant.

Semantics is commonly thought of as the "meaning" of symbols, of the manifold consequences which flow from their use. These consequences will be what effects the use of the symbol has on the world at large, since those consequences relating to other symbols are syntactic in nature. In its broadest sense, semantics can be seen as the relation of symbols to (nonsymbolic) events and processes. Semantics arise when a symbol is placed in some stable relation with the material world. Interpretation of words by human beings are semantic acts, as are acts of perception in organisms, as are the acts of measurement in physics. The direction of the relation need not be from the world to the symbol, but can be the symbol's manifold effects on the world. A switch turned on or off can launch rockets, start a conveyor belt, de-fibrillate a heart, or heat a room, depending upon its particular connection to the rest of the world.

In many cases the semantics of a symbol can be distinguished according to the directionality of the symbol-world relation. If the world determines the symbolic outcome, then the relation could be called one of *measurement*. Similarly, if the directionality flows from the symbol to an effect in the world, the relation could be called one of *control*. The function of syntactic processes is to allow the disengagement of these two types of relations so as to enable their flexible coordination. In effect, this allows for a symbol to have a unidirectional semantics. One functional constraint is simpler to fulfill than two. If the same symbol had to carry both measurement and control semantics rather than one or the other, then no coordinative flexibility and no syntactic relations would be possible.

The semantics of a symbol (here, as opposed to a token, a "sign") resists complete description.

> What is frequently appreciated in many so-called symbols is exactly their vagueness, their openness, their fruitful ineffectiveness to express a 'final' meaning, so that with symbols and by symbols one indicates what is always *beyond* one's reach. (Eco, 1984, p.130)

In general our knowledge of the world is incomplete, hence our knowledge of the relation between the symbol and the world is also incomplete. Our models are always limited by the finite (and small) number of observables which are available to us. Within some very limited and specialized contexts, such as within formal systems and computer programs, which are constructed and defined so that only a small number of observables are needed to effectively and exhaustively encode the context, we *can* know all of the relevant properties of the context. Once we have all the relevant properties of the domain, the relations between these properties can by described by logical rules operating on symbols. What were semantic relations become syntactic ones.

As a result, in computer science "semantics" generally refers to the function of a symbol within a particular computational context. Since the computer acts only on its own programmed rules and not on any symbol-environment relations, its symbols carry no inherent real world semantics. By redefining "semantics" to mean computational, logical consequences, the original concept is altered to resemble a species of syntax.

The current trend of the "syntactization of semantics" began in the 1930's with the "logical semantics" of Carnap and the "model-theoretic" semantics of Tarski (see general discussion in Lyons, 1977, chapter 6; see contemporary critique in Lakoff, 1987). This syntactization is accomplished by completely encoding the world, so that the symbol is seen in relation to a completely logical-symbolic structure. In true platonic fashion, sets of "possible worlds" and "world-states" can be postulated without having to specify any specific sets of observables and without having to verify any truth values with respect to the external world. Because of the limited number of observables any realizable model has, there is always an unencoded world external to the encoded realm. Consequently, there is always the possibility for "external semantics," although in model-theoretic accounts these seem to be ignored in favor of the "internal" or "logical" semantics. Bertrand Russell noted in this trend away from linkages with the sensible world a gradual slide into "perdition":

> Plato, who was interested in astronomy solely as a body of laws, wished it to be wholly divorced from sense; those who were interested in the actual heavenly bodies that happen to exist would, he said, be punished in the next incarnation by being birds. This point of view is not nowadays adopted by men of science, but it, or something like it, is to be found in the works of Carnap and some other logical positivists. They are not, I think, conscious of holding any such opinion, and would vehemently repudiate it; but absorption in words, as opposed to what they mean, has exposed them to Platonic temptation, and led them down strange paths toward perdition or what an empiricist must consider such. Astronomy is not a *merely* a collection of words and sentences; it is a collection of words and sentences chosen, from others that were linguistically just as good, because they described a world connected with sensible experience. So long as sensible experience is ignored, no reason appears for concerning ourselves with a large body having just so many planets at just such distances from it. And the sentences in which sensible experience breaks in are such as "that is the sun." (Russell, 1948, p.262)

Along with the slide away from real world observables, there is a predilection for examples taken from purely formal domains, such as number theory, where messy questions involving external semantics and choice of observables never arise.

Contemporary computer science and artificial intelligence have followed suit by gradually substituting "internal semantics" for external, real world semantics, manifesting precisely this tendency to "express all truth, all meaning, all sense, by means of syntax," thereby minimizing or obliterating completely the relation of symbols to the external world.

> Syntax, of course, is a linguistic concept originally. It pertained to the study of patterns of arrangement of letters within words, words within sentences, sentences within paragraphs, etc. It dealt with what was true or meaningful in a language solely by virtue of form and pattern, independent of any other considerations. It stood thereby in contrast to semantics: the study of meaning or signification. It would seem obvious that, in the study of natural languages at any rate, these studies must be complementary: that both pattern and meaning enter significantly into the very idea of a language, and that neither could replace the other. But syntax appears objective and scientific, while semantics involves conventions and subjectivities that seem to generate hosts of perhaps needless obscurities and ambiguities. Hence in a sense it has appeared that the more syntax and the less semantics in a language, the better it must be. In our age we have passed to the obvious limit and attempted to construct and use languages consisting of syntax alone. (Rosen, 1987, p.1)

Pragmatics. Pragmatics concerns the function of language, its usefulness to the user. While the semantics of a given word concerns its relations to the world at large, the pragmatics of the word concerns what functions the word enables, why that particular word is useful in a given context. Pragmatics is concerned with questions of why certain meanings are useful, how they fit into human purposes. For example, while the semantics of the various words the Eskimos use for what we generically call "snow" are the linkages between the words and the respective weather/ground conditions, the pragmatics of such words involve why these distinctions are useful to them in coping with the Arctic. While syntactics deals with necessary, rule-governed relations, and semantics deals with

contingent, interaction-dependent ones, pragmatics deals with intentional, goal-related, teleological ones.

Irreducibility. One can see three of Aristotle's causes in these types of relations: syntax as formal cause, semantics as efficient cause, and pragmatics as final cause (cf. Rosen, 1986). The three aspects of language are irreducible and orthogonal with respect to each other because the necessary, the contingent, and the intentional cannot be fully implemented by the others.

To destroy the syntax/semantics distinction would entail that the context be completely encoded, syntacticized, which is a very dubious proposition outside very restricted formal realms. On a different level, however, for us to distinguish the signs, we must make use of measurement processes which implement semantic relations between us as observers and the formal system being observed. To destroy the syntax-semantic distinction on the level of the observer is the same question as subsuming the description of the operation of measuring device into the description of the system being measured. It cannot be done without infinite regress (von Neumann, 1956). This is related to the measurement problem in physics, which will be discussed more in chapter 4.

To destroy the semantics/pragmatics distinction would entail that all evaluative criteria (i.e., function, utility, purpose, measure of performance, survival value) be subsumed into the linkage of symbols to the world. But such a reduction would mean collapsing linkages of symbols to functions in the world to linkages of symbols to structures in the world. Again, while some reduction could be accomplished within artificially limited contexts, complete reduction of functional to structural relations seems unattainable for most real world situations.

Syntactic, semantic and pragmatic relations for DNA are illustrated in figure 2.1. For DNA in a biological organism, syntactic relations would involve transcription-translation rules. DNA semantics would involve the relation of a particular sequence to the three dimensional protein structure it codes for and the relation of this structure to other biomolecules. DNA pragmatics would involve the relation of that DNA sequence and its constructional ramifications to the survival of the organism. These three categories , which biologists usually call the genetic, the phenotypic, and the selective properties of life are essential requirements for evolution. In effect, these categories define the organism as a semiotic entity.

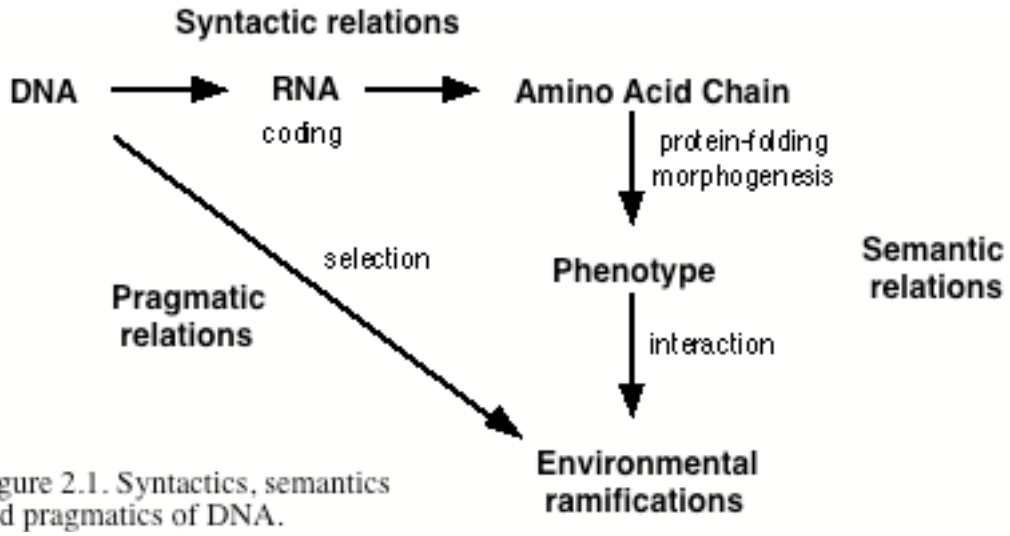

Figure 2.1. Syntactics, semantics and pragmatics of DNA.

Symbols and order

The way one perceives order in the world affects one's theory of symbols in relation to that order. If the order is definite and law-like and has only one true description then the relationship of the description to the underlying order must be a necessary one. If the order is indefinite and arbitrary and

multiple descriptions can be found to be useful, then the relationship of the description to the underlying order is a contingent one, contingent upon one's particular purposes.

The primary divide between means of viewing the role of symbols concerns theories of signification vs. theories of reference.

> Signification (that is, meaning) says what a thing is, and in this sense it is a function performed also by single terms; in the act of reference one says, on the contrary, that a thing is, and in this sense reference is a function performed only by complete sentences. (Eco, 1984, p.23)

Signification is semantic, contingent, empirical while reference is syntactic, necessary, logical. In a deep sense the split is over the nature of symbols relative to order in the world, whether they are necessary reflections of an underlying, rational, necessary order or whether they are contingent consequences of an as of yet undefined one.

This split consequently manifests itself in all the important philosophical discourses, in the theory of language use and acquisition, in the theories of truth, in the status of mathematical objects, in models of the mind, and in scientific methodologies. Rationalistic philosophy, from Leibnitz to Frege, Russell, Carnap and Tarski has always emphasized syntactic relations, while empiricist tendencies, from the British empiricists on, have emphasized the semantic aspects of symbols (Morris, 1946, pp. 286-287). The American pragmatists, Pierce, James, Dewey, and Morris also held the pragmatic aspects of symbols to be of the same importance as both syntactic and semantic relations. Consequently, relative to these categories, the pragmatist conception encompasses the categories of both rationalism and empiricism. The profound difference between pragmatism on one hand, and both rationalism and empiricism on the other, is that pragmatism sees the three aspects of syntax, semantics, and pragmatics as irreducible. In the contemporary analysis of symbol systems rationalism and pragmatism seem to prevail (Eco, 1984, p.1). We thus have two general theories of the sign, one in which the sign is a *logical* correlation between signifier and signified (symbol as reference) and one which focuses on the *purposes and origins* of signification (semiosis). The first is founded on logical necessity, while the second is based on the interrelationship between logical necessity, empirical contingency, and pragmatic intentionality.

Symbol as reference.

The concept of designation or reference appears in many discourses about symbols and goes back to Plato. A symbol is an object or figure which stands for something else, as in the way the word "chair" stands for the material objects one sits on. The symbol "points to" or "stands for" that to which it refers. This is the concept of symbolic reference or representation, and this concept pervades computer science, cognitive psychology, and much of analytic philosophy.

Both rationalist philosophies of mind and realist philosophies of science adopt this position. In this view the mind is regarded as a mirror of nature, symbols in the mind correspond to definable, identifiable objects in an objectively knowable real world. When relations between symbols in representations of the world correspond to the way that things really are, these relations are said to be "true." This worldview assumes that both the signifier and the signified are in some sense definite entities. Unfortunately, the only means by which the "state of the real world" can be assessed is through other symbols (via other measurements) whose relations to the real world are themselves no better known. There are no unmediated channels of access to an underlying Reality, and hence no means of absolutely determining the truth value of a proposition independent of all others.

This concept of symbolic reference is only useful if one has an unambiguous agreement on what is being referred. The clean, well-defined picture of reference only holds when all entities have unambiguous properties, are accessible to all observers, and have definite, stable, code-like relationships. It is no accident that this conception of symbol is closely tied with the mechanistic view of

the world exemplified by "classical" physics, where the world itself is assumed to be exhaustively describable and deterministic.

There exist domains, such as mathematics, which have been constructed so that all the relevant properties of the entities in the domain are completely accessible to all observers in an unproblematic way. It is understood by the observers and their common definition of the context what these properties are and how to observe them so as to achieve agreement and replicability across observers. In these completely encoded, formal realms, notions of symbolic reference are useful because it is always possible to point to the well-defined correlate properties of any representation. Thus for formal, mathematical entities, computer programs, or "model-theoretic," "possible worlds," it *can* make sense to talk about representations and the objects they represent.

For "natural" domains in which the definition of the domain itself is not itself constructed to be unambiguous, however, the concept of representation is of dubious value. When applied to biological contexts, symbolic reference breaks down completely as an explanatory principle. If we ask what a particular sequence of DNA "represents," "corresponds with," or "points to" in the organism's environment no answer will be immediately forthcoming because the complete characterization of biological environments is itself problematic. And because biological environments are not well characterized, we have no exhaustive description through which biological symbols could be related to features of the environment. Thus, any attempt to discern what a DNA sequence in a particular organism refers to in the outside world will depend upon how we choose to describe the world at large.

It might be argued that a DNA sequence "represents" or "points to" the protein that it codes for, but even this usage of the concept is fraught with difficulty. Even if DNA is thought of as representing the protein that it codes for, the actual tertiary structure of the protein itself is by no means completely defined. Biological structures have multiple functions and effects on the environment, and each function may not be precisely known. The problem of exhaustively characterizing the organism's phenotype parallels that of exhaustively characterizing its environment.

Symbol and purpose

The major alternative to the representational account of the role of symbols sees symbols as instruments which are useful for particular purposes within particular contexts rather than logical linkages with the external world. Rather than representing the world, symbols become vehicles for reliable communication and effective action. Symbols shift from bearers of "truth values" to instructions having various context dependent effects and utilities. This is a more appropriate conception for describing what goes on in biological organisms.

> Human language must have been produced by evolution. Evolutionary forces--and that means natural selection, the only evolutionary force we know of--could have produced it only if it produced effects. And a coding of natural phenomena into symbols, into a language, could only produce effects if the basic character of the language is to be imperative, not indicative; to express, in symbolic form, commands, instructions, programmes--not statements.
>
> This is, of course, the grossest heresy against the orthodoxy with which people have attempted to indoctrinate me since I was a young man. The whole gist of language, I was told, is concerned with statements--analytic or empirical. A command cannot be asserted or denied, and, according to Karnap for instance, is therefore 'without sense'. Present day linguistic philosophers have relented to some extent from the extreme Fundamentalism of Logical Positivism, but, in my view, not nearly enough. To the natural selective forces which bring about evolution--of language, as of anything else--it is precisely commands, producing effects, which do have sense, and it is statements about facts--atomic or otherwise--which are without significance, or meaning. (Waddington, 1972, p.288)

In biological organisms, almost all symbols have the function of directing or modulating an action of some sort. The symbolic type becomes a means of deciding or switching between action alternatives. According to this view, a symbol does not have to *represent* anything to effectively coordinate action, and the meaning of the symbol itself is completely related to the effects of its use, its material

consequences. Symbols, having an arbitrary relation to those things to which they are connected in consequence, come to acquire meaning only through their use. Through their use, the relation of the symbol to the world becomes selected according to its utility. Thus, although symbols are governed by the same material laws as everything else, their relation to the world is contingent upon their history and their purposes.

Instructions can have either syntactic or semantic aspects, depending upon whether one is considering their effects on other symbols or on the world at large. DNA, seen as instruction, has both depending upon which effects one chooses to consider. DNA directs the formation of amino acid chains which then fold up into three dimensional structures with enzymatic functions. The symbolic aspects of DNA can be seen as a very special kind of physical constraint, which recognizes each codon as a discrete type and provides the appropriate amino acid. A simple rule can be written to describe the relationship between DNA codons, RNA anticodons, and the resulting amino acid which will be attached to the growing protein chain at the ribosome. This process can therefore be seen as syntactic in character. Furthermore, it can be seen as the action of some very localized stereochemical structural relationships (e.g. t-RNAs) which are implementing localized rules. A DNA codon does not become a particular protein sequence in the absence of a large array of very specialized enzymes.

After the protein chain is synthesized, the protein folds up into a three-dimensional native conformation in most cases having some sort of functional specificity. This process is not readily described in terms of simple rules, but appears to us as a complex, analog process in which the inherent properties of the amino acids themselves (e.g. hydrophobicity, electrostatic charge distributions, flexibility) take part. These are properties which apply to the amino acids outside any local context; they resemble physical laws. Even the simplest models of protein folding do not suggest the action of simple localized folding rules; they rely on attempting, however crudely and indirectly, to model the law-like properties inherent in the amino acids themselves through various kinds of mathematical constraints (e.g. Cariani and Goel, 1985). While we cannot identify a rule-governed, syntactic relation between the DNA and the folded protein, there is a stable, structural connection between the two, one which has manifold effects both inside and outside the cell. The relation of the DNA instruction to the folded protein is thus a semantic one, not unlike that between the symbolic instructions to a numerically-driven machine tool and the resulting physical motions it produces. More familiarly, it is the relation between the motions induced by the instruction, "Please pass the salt" and the instruction itself.

The instructional capabilities of symbols make possible communication, language and coordinated activity, hence multicellularity on an organismic scale and community and culture on a social scale:

> Combined activity happens among human beings; but when nothing else happens it passes as inevitably into some other mode of interconnected activity as does the interplay of iron and oxygen of water....Only when there exist *signs* or *symbols* of activities and of their outcome can the flux be viewed, as from without, be arrested for consideration and esteem and be regulated. Lightning strikes and rives a tree or rock, and the resulting fragments take up and continue the process of interaction, and so on and on. But when phases of the process are represented by signs, a new medium is interposed. As symbols are related to one another, the important relations of a course of events are recorded, and are preserved as meanings. Recollection and foresight are possible; the new medium facilitates calculation, planning, and a new kind of action which intervenes in what happens to direct its course in the interest of what is foreseen and desired.
>
> Symbols in turn depend upon and promote communication. The results of conjoint experience are considered and transmitted. Events cannot be passed from one to another , but meanings may be shared by means of signs. Wants and impulses are then attached to common meanings. They are thereby transformed into desires and purposes, which, since they implicate a common or mutually understood meaning, present new ties, converting a conjoint activity into a community of interest and endeavor. Thus there is generated what, metaphorically, may be termed a general will and social consciousness: desire and choice on the part of individuals in behalf of activities, that, by means of symbols, are communicable and shared by all concerned. A community thus presents an order of energies transmuted into one of meanings which are appreciated and mutually referred by each to every other on the part of those engaged in combined action. (Dewey, 1928, p.471)

Through language, cells and individual human beings can be coordinated in such a way that a larger, more complex organization of conjoint activities becomes possible.

The evolutionary advantages of this kind of complexification are enormous. Through language, linkages can be constructed between the experiences of different individuals, thereby amplifying the range of possible epistemic connections each individual can have with the world at large. These linkages in effect implement new semantic relations, altering the relation of symbols to world by syntactic couplings of the previously unconnected percepts and actions of separate individuals. These linkages can also apply to organisms, devices, and even simple tools as well as other human beings. Our eyeglasses, our automobiles, our walking sticks effectively become parts of our bodies. The existence of this process of syntactically coupling ourselves to external measuring instruments and effectors is one basis for the expansion of our set of semantic relations with the world. We will see this process at work again in chapter 10 when we examine devices which create their own semantic relations and in chapter 12 when we consider this phenomenon in the context of emergence. The next chapter will consider the role of symbols in biological systems utilizing the general semiotic concepts and categories introduced in this one.

Chapter 3 The biology of symbols

> The question is whether information is to be treated in biology as just another physical variable, or as the characteristic and exclusive aspect of living systems and their artifacts that distinguish living systems from all other physical systems. (Pattee, 1979, p.218)

> To a biologist, therefore, a language is a set of symbols, organized by some sort of generative grammar, which makes possible the conveyance of (more or less) precise commands for actions to produce effects on the surroundings of the emitting and the recipient entities....the basic sentences in language are programmes, not statements. And it is language in this sense--not as a mere vehicle of vacuous information--that I suggest may become a paradigm for the General Theory of Biology. (Waddington, 1972, pp.288-289)

Symbols in biological systems

The most striking aspect of symbols in biological contexts is their ubiquity. Virtually all physical systems which we intuitively recognize as living engage in processes of encoding, decoding, and symbol manipulation. *We cannot understand the organization of living things without understanding the role symbols play in biological organization.*

The converse is also true, that virtually all symbols are associated with biological organisms, whether at a cellular or at a social level. *We cannot understand symbols until we understand their role in the organization of life.*

Not only are symbols found in every DNA-based organism, but within many multicellular organisms various other kinds of symbols also operate on different organizational levels. At the level of intracellular organization, DNA can be seen as a coding system which instructs the formation of virtually all structural proteins and enzymes. Above the level of intracellular codes, there are hormones and neurotransmitters, along with their associated receptors, which perform the function of switching recipient cells, tissues and organs into alternative behavioral modes. Above the organismic level, on the level of communication between individual organisms, the symbol primitives can be sounds (warning cries, human language), light (fireflies), posture (antelope, prairie dogs), or any number of other physical events that organisms can both produce and detect (see Bonner, 1980 for examples and an account of their evolution).

A systematic, multi-level list of biological symbolic activity can be found in James Miller's (1978) massive, functionalist taxonomy of living systems, one which extends from organelles to societies. (Miller, 1978, under "3.3...Systems which process information" for each level).

One of the major difficulties in defining biological organisms in terms of symbols has been the lack of any coherent theory of symbols or of information in biological systems. What, exactly are symbols, such that we can recognize when they are operating in natural contexts? Without a clear theory of symbols, a biological semiotics, it is arguable that the subject matter of biology will remain undefined:

> At root, theoretical biology is concerned with only one great question: What is it about certain material systems that confers upon them the characteristics of life, which makes them living beings? All other problems of biology, both theoretical and practical, are collateral or subordinate to this central question.
>
> It is a significant fact that, despite generations of trying, there is as yet no list of tests, characteristics, or criteria we can apply to a given material system that can decide whether that system is an organism or not. Stated another way, the decision as to whether a given system is an organism is entirely a subjective, intuitive one, based upon criteria that have so far resisted formalization, or even articulation. Thus, from a strictly rigorous point of view, the subject matter of biology is *undefined* ; it is based entirely on an informal consensus essentially akin to pattern recognition, but that consensus is one that we all share to a startling degree. (Rosen, 1985, pp.166-167)

The task of theoretical biology is to clarify and deepen that consensus, not to uncover some "essence of life." Two thoughts by Neils Bohr come to mind:

> We meet here [in quantum mechanics] in a new light the old truth that in our description of nature the purpose is not to disclose the real essence of the phenomena but only to track down, so far as it is possible, relations between the manifold aspects of our experience. (Bohr, 1934, p.18)

> I am quite prepared to talk of the spiritual life of an electronic computer; to say that is is considering or that it is in a bad mood. What really matters is the unambiguous description of its behavior, which is what we observe. The question as to whether the machine *really* feels, or whatever it merely looks as though it did, is absolutely as meaningless as to ask whether light is 'in reality' waves or particles. We must never forget that 'reality' too is a human word, just like 'wave' or 'consciousness.' Our task is to learn to use these words correctly--that is, unambiguously and consistently. (from Wheeler and Zurek, 1983, p.5)

Several possible means of describing the role of symbols in biological contexts are available. Each illuminates a separate set of issues, each is a partial theory. None forms a comprehensive, consistent account of the full gamut of biological symbolic activity. These theories include biological information theories, biological thermodynamic theories, functionalist accounts of the various roles symbols play, computationalist models of symbol-processing, von Neumann's theory of self-reproducing automata, and cognitive evolutionary perspectives.

Biological information theory focuses on biological symbols as information channels, while biological thermodynamics provides a theory for analyzing why symbols as dissipative systems can be structurally stable. The functionalist account involves the general usefulness of symbols for the functioning and stability of the organism. The computationalist viewpoint sees biological symbols as analogous to those manipulated by computers. These viewpoints are evaluated and criticized in the sections below. The cognitive interactionist perspective, which is adopted in this dissertation, emphasizes the advantages for the organism of cognitive connections with the environment.

Biological information theory and biological symbols

Biological symbols can be analyzed in terms of the statistical communication theory of Shannon and Weaver (McKay, 1969; see Singh, 1966, and Pierce, 1961, for general introductions), often called "information theory."

The theory involves a signal source, a transmission channel, and a signal receiver. It is assumed that a finite set of distinguishable messages are sent through the channel. The information content of a particular message is related to the probability distribution of the entire set of possible messages. The information content of an improbable message is higher than a more probable one because it results in a greater reduction in uncertainty.

DNA can be seen as an information channel spanning generations. The signal source is the original genetic sequence in the reproducing parent(s), the receivers are the sequences in the offspring, and the channel is the process of making multiple copies. Information theory can be used to assess the reliability, efficiency, and carrying capacity of the genetic transmission process.

Information theory concerns itself with a completely self-contained domain of previously encoded message possibilities. It does not explicitly deal with the process of encoding or the generation of new encodings. Even though the transmission of these symbolic messages can be a stochastic process and therefore not rule-governed, the bounds of the theory lie completely within syntactics, the relation of symbols to other symbols. However, the most important relations for symbols in biological contexts lie within the semantic and pragmatic realms. In information theoretic terms, semantics involve how the message is related to the world, what is its meaning, while pragmatics involve why the message was chosen for transmission, what is its utility. These considerations lie outside the communication channel; consequently, they are not usually a part of the explanatory domain of the theory.

Because DNA sequences are both selected sequences of nucleotide bases and instructions coordinating the construction of the organism, Goodwin has argued that "knowledge" needs to be

distinguished from "information" in biological systems. While information refers to syntactic constraints, knowledge encompasses also semantic and pragmatic aspects:

> The technical definition of information involves only selection (e.g. specifying one out of the set of possibilities), but says nothing about meaning, which I take always to involve activity in real space-time. Thus knowledge differs from information in that it not only involves selection of alternative possibilities, but also includes instruction for action which, operating in a particular context, conveys meaning. (Goodwin, 1978, p.120)

In this regard, Donald McKay (1969, chapter 7) attempted in the early 1950's to develop an account of meaning in information-theoretic terms. His solution was to ground semantics behavioristically, by defining meaning as the behavioral effect a message evokes upon its receiver. This effect is mathematically defined in terms of changes to the receiver's conditional probability matrix. Given that the receiver has gotten a given message, what is the probability that s/he will embark upon a particular course of action? This conditional probability matrix, however, is in terms of a fixed set of behavioral states, each corresponding to a distinct course of action, for the receiver. This is fine as long as it is understood that this definition of meaning is relative to a particular model of the receiver and, due to the impoverished nature of that model, does not exhaustively describe the effects of the message on the receiver. If, on the other hand, one assumes that the effects of a given message can be fully encoded, then the information theory begins to resemble the model-theoretic semantics of Bar-Hillel and Carnap (McKay, 1969, pp.81-83). All aspects of the receiver not in the model of the receiver are subsequently lost, and gradually the existence of these unencoded factors disappears completely from consideration. Once the semantic effects are completely reduced to a finite set of discrete alternatives, then all semantics becomes conceptualized as operating within those alternatives. New states and semantic primitives are thereby eliminated from the theory. All that remains is the building up of combinations of pre-existing primitives (Lilienfeld, 1978, pp. 80-88). Information theory thus comes to obscure the basic problem of the origin of new semantic primitives.

Rather than dealing with the problem of how new states come to be defined, that is, passing from undefinition to definition, information theory substituted stochasticity of state transitions between existing states. Thus randomness came to be associated with novelty and information generation, rather than seeing these as being necessarily associated with new measurement processes.

Information theory was intimately associated with early cybernetic ideas and therefore has many connections with the perspective developed here. If the formalistic notion of an exhaustive reduction of semantics to syntax is dropped and the problem of the origin of new states is not avoided altogether, then the theory would be more useful for the present purposes of designing devices which create new semantic relations.

While information theory, as currently conceived, cannot explain the origins of new semantic primitives in messages sent through a channel, it *is* useful in quantifying the number of distinctions which are communicable through syntactic means via a particular channel. In this respect the information-theoretic analysis of the *relative* information transmission capacity of DNA in different species is quite useful (Gatlin, 1972). This is possible because the set of alternative message possibilities is known given that all DNA messages must be finite strings composed of four primitive symbols, the four nucleotide base types. In this case the "states" of the system are relatively unproblematic because of the discrete symbolic nature of the DNA bases. It is much more problematic to ascribe information content to particular phenotypic structures when we have little or no idea of what the range of possible structures might be.

The syntactic aspects of biological symbols increase in importance as the number of symbols increases and as their interplay becomes more flexible. As one examines more complex organisms with more flexible behavior patterns and communicatory capacities, syntactic constraints can play a large role in increasing the informational capacity of the channel.

Thermodynamic theories of biological information

Symbols are dissipative structures, basins of attraction, owing their stability and their reliability of behavior on the dissipation of energy. The action of symbols therefore involve a concomitant export of entropy to the environs of the organism. Only particular kinds of thermodynamic structures, therefore, can serve as the material substrates for symbolic activity.

Because biological systems are continually dissipating energy, they must exist as thermodynamically "open" systems, having external sources of energy (Bertalanffy, 1968, chapter 6). Biological thermodynamics analyzes how such systems can persist over long periods of time. But energetic considerations are not enough for adequate explanation:

> It is a truism that living organisms must take up, utilize, and dissipate energy or they will die. It is also true that no living organism can survive whose minimum energy needs exceed available supplies. From this, one might then assume that energy flowing through the surrounding environment creates the boundary conditions for organisms. But we would assert that the flow of energy cannot explain the structure of living organisms. Energy is modified or differentially utilized by organisms, and that modification or use is determined by properties intrinsic to the organism....Energy flows do not provide an explanation for why there are organisms, why organisms vary, or why there are different species. (Brooks & Wiley, 1986, p. 37)

Thermodynamic theorists consequently believe that the explanations of these differences must be explained in entropic and informational terms. Many of the current ideas are built upon the work of Ilya Prigogine (1980; Prigogine & Stengers, 1984) on irreversible processes and far-from-equilibrium systems. There is a connection between the thermodynamic view and the biological information-theoretic view, in that both the concepts of entropy and information deal with sets of accessible states, observable macrostates, and probabilities of the natural system being particular states. These involve issues of order and complexity, of subjective and objective conceptions. Misunderstandings and tacit cross-overs between "observer-centered" and "reality-centered" interpretive frameworks, starting with the Gibb's Paradox, have greatly complicated the definition of quantities like "information," "entropy," "order," "complexity." All are functions of the number of states of a system, but very seldom are either precise state definitions or their means of measurement in some natural system given explicitly. There is a discourse in progress attempting to explicate these relations (e.g., in Weber *et al.*, 1988), but it is one which is too complex and/or entropic to be disentangled here. We have yet to see a clear consensus emerge on the relations between thermodynamics , dynamical theory, and the emergence of symbols, but the problem is being actively addressed (e.g. Swenson, 1989).

Where information theory is a theory of syntactic transmission capacity, biological thermodynamics is first and foremost a theory of structural stability. As a theory of stability, its focus is on the internal structural conditions of persistence rather than on performance within an external environment:

> And why do particular organisms look the way they do? Because each is the product of a unique history in which intrinsic constraints restricted the number of kinds of variants that were not adequate for survival. Biological evolution is not a teleological process, nor is it a process that requires us to postulate better adapted variants occur randomly and are 'selected' because of their functional efficiency in a given environment. Rather the most urgent property of living systems as entropy systems is historically constrained structural evolution regardless of the environment. Evolution is the survival of the adequate, not of the most fit. (Brooks & Wiley, 1986, p. 71)

In this view the inherent tendency of open, nonequilibrium systems to self-complexify is a more important evolutionary driver than natural selection:

> At the heart of this proposal is a notion of 'information dissipation' in populations that have built up enough variation to threaten gene linkages in and between populations. Brooks and Wiley are inclined to treat this process as an inference from a single, overarching formulation of the second law, expressed in information-theoretic terms ... the

> law-like status of this physical-informational principle leads to the *necessity* of phylogenetic branching, and to the suggestion that branching can be treated as an informational analogue of bifurcations in the theory of nonequilibrium thermodynamical systems. Because the object on which this process works is the species ... phylogenetic branching is not, properly speaking, caused by processes going on at levels below the species, or indeed, because of interactions between species and their environment. Rather it is an inevitable and autonomous consequence of the working of the second law and would not otherwise occur.
>
> This does not, however, mean that natural selection plays no role. On Brooks and Wiley's account, natural selection acts primarily as a negative and pruning factor controlling the rate at which this branching process occurs. (Depew & Weber, 1988, p. 341)

Related to internal thermodynamical constraints are other kinds of organizational constraints which channel evolutionary trajectories (Frazzetta, 1975; Riedl, 1978). Simulations of the evolution of genetic regulatory networks by Stuart Kaufman also point to universal organizational properties of large interconnected networks which may channel or even resist external selection pressures (Kaufman, 1985).

If biological thermodynamics eventually succeeds in coping with the complexity of biological systems, the result would place constraints on which possible structures were stable, rather than predicting which ones actually arise. This would be useful in delimiting evolutionary possibilities, in explaining why we see the particular types of biological forms we see. We would also gain important insights into the thermodynamic structures which are necessary for the formation of new symbols and their semantic linkages.

However, such an accomplishment would not obviate the need for symbols in our theories of evolution. The means of stabilizing and switching between alternate structures remain primarily symbolic ones, and the symbolic level of description retains its usefulness even if an underlying (thermo)dynamical account is feasible. The structures formed by thermodynamic bifurcations must be replicable, inheritable from generation to generation. As will be discussed in the next section, it is far easier to create new structures which use genes as boundary conditions than ones which must rely on cytoplasmic inheritance, even via interaction with large catalytic networks (e.g. as in cycle formation in catalytic nets, as in Kauffman, 1968). The relative rate of evolutionary complexification before and after the appearance of DNA is perhaps evidence for this hypothesis.

The functions of symbols in biological contexts

Functional analysis of biological symbols involves identifying particular functions or roles that symbols carry out such that the organism persists. Symbols play a number of biological roles, primarily as stabilizers of very complex, yet replicable, spontaneous orders. The use of symbols allows for a higher degrees of stability and complexity than would otherwise be possible

Islands of stability. First and foremost, symbols represent the means of stabilizing relations between parts of an ever-changing system. Symbols can be seen as the more stable parts of the system which have the capacity to retain their identity despite constant change in the rest of the system. DNA is one of the most stable macromolecules in the cell, as evidenced by the relatively high temperatures to which it can be subjected without breakage and consequent loss of information.

Because of their stability symbols become the boundary conditions within which the more transient biochemical dynamics of protein-folding and catalysis occur. The entire genome functions as a very complex boundary condition for the interactions of the enzymes, substrates, and structural molecules of the cell. When the boundary conditions are stabilized, more complex dynamics can be realized and reliably controlled.

The persistence of memory. Symbols through their relative stability and their discrete interpretation implement a memory function for the system. Memory entails the relative stability of a particular physical configuration. The expression of a single gene causes a particular protein to be manufactured, altering the dynamics of the cell in some way by switching it into another stable interaction mode.

Specific dynamical states can be recovered, the system can be reset by the action of symbols. Past events can also be encoded into symbolic form and utilized as "information about the world" at a later time.

Rule construction and execution. Symbols allow for discrete biological "states" and rule-governed switching between these states which can be made independent of the underlying molecular dynamics. This allows for consistent, controllable responses in a variety of domains.

In morphogenesis, rule-execution enables the same "switchings" between complex dynamical states to be carried out generation after generation. What makes these switchings different from those of a computer is that the resulting complex physical dynamics are not immediately reinterpreted symbolically.

In reacting to its environment, rule-execution enables behavioral consistency. The organism can thus carry out the same actions in response to a given stimulus. Behavioral consistency enables testing of behavioral alternatives and mechanisms for the optimization of behavior. Without behavioral consistency, optimal responses cannot be reliably re-created. Without rule-governed control, perturbations will eventually erode the performance of the system.

Flexibility with consistency. The arbitrariness of the relation between the symbolic "states" and the dynamics allows for the possibility of the flexible switching. All combinations of input "states" and output "states" become possible.

An example of this flexible switching would be the mapping between RNA anti-codons and the amino acids they code for. Each anti-codon is a functional input "state" and each possible amino acid is a functional output "state." Alternative t-RNA molecules could implement all the possible anti-codon-amino acid combinations.

Coordination. Another example of flexible switching lies in the coordination of sensory inputs with behavioral alternatives, the mapping of stimulus to response. Assuming that some combination is optimal for the organism, the greater the flexibility of the switching apparatus, the greater the likelihood that the organism can implement the optimal combination.

The most fundamental function of the nervous system lies in its ability to coordinate the actions of different parts of the organism in order to achieve higher functions that would otherwise be unachievable.

Like the generalized encoding of different functions by DNA and the advantages that a generalized autocatalytic system possesses, the nervous system becomes a generalized coordinator of actions. Nearly all organs and tissues have some nervous system modulation.

On the level of the coordination of behavior by a nervous system, this computation-universality means that the system has the capacity to match any output (action) contingent upon any input (percept). This flexibility of input-output mapping creates the greatest possible number of realizable contingencies, thereby increasing the likelihood that function will be improved.

Communication. On the inter-organismic level, communication allows for the coordination of individual organisms, thereby allowing an individual animal to amplify its own senses and effectors by acting in unison with others (Bonner, 1976). Any medium in which the organism can produce changes which it itself can reliably distinguish can function as the basis of a communication system. Any medium in which the produced and distinguished changes can be placed into discrete categories can function as the basis of a discrete, symbolic, communications system.

Repairability. The discreteness of symbols, that they are recognized as belonging to one of a finite number of discrete types, also allows for their repair. Several DNA repair systems (e.g. excision repair) can restore altered sequences to their original form, thereby further stabilizing the genome from mutational perturbations. Redundancy in the coding provides for greater reliability in the face of mutational events.

Reliable heritability. The symbolic nature of DNA means that symbolically identical copies can be made. Errors and perturbations can be reduced by adding redundancy in the form of multiple copies of

each gene. Mutational errors can be coped with by increasing the number of offspring; the greater the ratio of offspring to parents, the more likely a given sequence will be propagated completely intact in at least one individual. Were this an analog process, small perturbations would not be damped out by the symbol system and would propagate from generation to generation.

Replicability of complex dynamical states. Symbols make complex ontogenetic development possible. The capacity for symbols to retain their identity or "state" means that they can be used to switch dynamical systems into particular modes. Morphogenesis is an example of a very complex set of switchings by genes which steer the developing organism through a very complex dynamical space. Without symbolic modulation it would be very improbable that approximately identical adult forms would be preserved from generation to generation.

Generalized autocatalysis. Symbols dramatically increase the rate of evolution, and this is the most likely reason why all extant living organisms use symbolic encoding.

Generalized encoding, as found in DNA-RNA-protein, allows for each genetic sequence to be self-catalyzing, since each genetic sequence can produce a product which affects the stability of the system as a whole. Once the apparatus for replicating the DNA, the transcription-translation enzymes, and the replication apparatus itself is in place, all other genetic sequences can "piggyback" on the sequences that have the autocatalytic function. This means that a protein need not be connected into another catalytic network in order to be replicated in the next generation. Proteins henceforth are not constrained by having both a reproductive function and another positive utility to the organism. The number of proteins with useful functions which will also propagate to succeeding generations consequently increases enormously. The ability to enhance stability of the organism and the ability to self-replicate are thus separated for gene-protein combinations.

Von Neumann recognized this ability of a self-reproducing system to produce other, nonreplicating objects:

> ...Let X be A+B+C+D, where D is any automaton [and A+B+C is a self-replicating combination of automata]. Then (A+B+C) + F(A+B+C+D) produces (A+B+C+D) + F(A+B+C+D) [the object itself in parentheses, e.g. (X), its construction description as a function, e.g. F(X)]. In other words, our constructing automaton is now of such a nature that in its normal operation it produces an object D as well as making a copy of itself. This is a normal function of an auto-reproductive organism: it creates byproducts in addition to reproducing itself. (von Neumann,1948, p.489)

Were there no generalized encoding, every component would have to catalyze its own formation. The alternative would be take part in a network in which each component catalyzes the formation of some other component in the network, such that the formation of all components is catalyzed. While this does not prevent such networks from arising, this requirement dramatically slows down the rate of appearance of new proteins and new functions.

With generalized encoding, complex sequences of amino acids can be strung together reliably, allowing for very complex folding patterns to be dependent upon specific primary sequences. Were the process not digital, the probability of replicating a protein of a specific sequence would be a remote possibility. Increasing the reliability of the protein construction process has the added advantage of decreasing the amount of energy and substrates wasted in synthetic errors.

Another advantage of generalized encoding is that virtually all DNA sequences encode for proteins. This means that favorable genetic errors are propagated. These mutational products may not have corresponding synthetic pathways, aside from genetic transcription-translation routes. Thus protein sequences which would not readily occur without encoding have a much higher probability of occurring as a result of encoding.

Evolvability of complex structures. The general argument in favor of digital or symbolic encodings for evolving complex structures was made by John von Neumann in his general and logical theory of automata (von Neumann, 1948). For complex structures to be reliably replicated from one generation to

the next, the noise level or the tendency for error in the formation of the structure must be reduced. Optimal complexity can thus be seen as a tradeoff the number of attainable states vs. the reliability of attaining particular states. More attainable states mean a greater likelihood that some of the added states have greater adaptive value. Greater reliability in attaining those states means a higher probability of attaining the higher performance levels associated with some of those states.

This is similar to the problem of increasing the number of representable symbolic states with minimal increase in inaccuracies of representation ("noise").

> *The role of the Digital Procedure in Reducing the Noise Level.* The important difference between the noise level of a digital machine ... and an analogy machine is not qualitative at all; it is quantitative. As pointed out above, the relative noise level of an analogy machine is never lower than 1 in 10^5, and in many cases as high as 1 in 10^2. In the 10-place decimal digit machine referred to above the relative noise level (due to round-off) is 1 part in 10^{10}. Thus the real importance of the digital procedure lies in its ability to reduce the computational noise level to an extent which is completely unobtainable by any other (analogy) procedure. In addition, further reduction of the noise level is increasingly difficult in an analogy mechanism, and increasingly easy in a digital one. In an analogy machine a precision of 1 in 10^3 is easy to achieve, 1 in 10^4 somewhat difficult, 1 in 10^5 very difficult; and 1 in 10^6 impossible, in the present state of technology. In a digital machine, the above precisions mean merely that one builds the machine of 3, 4, 5 and 6 decimal places, respectively. Here the transition from each stage to the next one gets actually easier. Increasing the 3-place machine (if anyone wished to build such a machine) to a 4-place machine is a 33 percent increase; going from 4 to 5 places, a 20 per cent increase; going from 5 to 6 places, a 17 per cent increase. Going from 10 to 11 places is only a 10 per cent increase. This is clearly an entirely different milieu, from the point of view of reduction of 'random noise,' from that of physical processes. It is here- and not in its practically ineffective absolute reliability --- that the importance of the digital procedure lies. (von Neumann, 1948, pp. 398-399.)

The functionalist account of biological symbol function is useful in examining how symbols enable various useful capabilities. One limitations of this kind of analysis, however, is that the functions need to be related in a systematic, coherent way for the entire organism to persist and evolve. This can be done by specifying a list of fixed functional categories which are necessary for the organism to persist (e.g. Miller, 1978). Alternatively coherence of function can be addressed by the dynamic interconnectedness of ongoing processes, as in the autopoiesis theory of Maturana and Varela (Varela, 1979; Maturana, 1981).

Biological computationalism

Computationalist theories of biological symbols start with the digital computer as their explanatory metaphor, and draw the conclusions that all of the biological functions listed above are, on a molecular level, syntactic, rule-governed processes.

> Aside from [Newtonian mechanics], we have one other source on which to draw in our attempt to come to terms with biological systems and other types or organizations. This source is our own technology; the fact that we routinely build organized systems to do things for us. Norbert Wiener remarked that our view of biological systems has always been heavily conditioned by our technology; in the 18th century, organisms were visualized as clockworks or automata, in the 19th century as engines, in the 20th as computers. Thus, if we are careful, we have at hand a class of systems whose organization we understand because we built them, and which we can use both to guide our intuition and also as a class of test systems to which we can apply any general theory of organism or organization which we may propose. But this must be used with care, as we shall see. (Rosen, 1974, p.167)

Although the newly forming field of artificial life covers both the "synthesis and simulation of living systems," computationalist assumptions seem dominant in most of the field. Because of importance of artificial life as a research domain and its anticipated effect on theoretical biological thought, an extended discussion and criticism of some major computationalist premises seems warranted.

Biological computationalism embodies the view that all biological processes, on one level or another, are literally symbol-processing operations. Each molecule is seen as a miniature computational

device which has a rule-governed state-transition behavior. The organism can be thus conceived as a large aggregate of interacting molecular automata. Consequently, there is *always* some formal system which is isomorphic to the structure of the biological organism.

There is a definite affinity for this view and the view that the entire universe is a gigantic computing machine (e.g. Toffoli, 1982), although in the wake of the discoveries and debates of quantum mechanics (in particular, von Neumann's mathematical demonstration that hidden-variable theories cannot exist (von Neumann, 1956; Bohm, 1952, p.390)), it is hard to see how either view can be held as more than a limited metaphor.

Biological computationalism holds atomist, mechanist, realist, and reductionist tenets. As a realist position, many computationalist claims are presented as articles of faith without any account of what the atomistic computational primitives of the (real) world might be or how they might be observed. If one level does not lend itself to exhaustive description via computational atoms, then deterministic "hidden variables" on the next sub-level can always be postulated.

These tendencies notwithstanding, there are a number of very positive aspects to the computationalist approach to biological organization employed by those doing artificial life simulations. Many important lessons *have* been learned from the failure of symbolically based, top-down artificial intelligence. These include a commitment to simulating underlying biological mechanisms rather than merely passing Turing tests for lifelike macro-behavior, a preference for evolutionary, bottom-up processes rather than static, top-down programming, a willingness to learn from "free running" computer simulations, and the sustained search for simple rules of interaction which are capable of generating behavior of high complexity. Questions of structural complexification and emergence are considered important and explicitly addressed, if not in the way some of us would prefer. Thus, some of the mistakes made by its predecessor, artificial intelligence, have been recognized and avoided.

Simulations vs. realizations. Unfortunately, many of underlying assumptions of traditional artificial intelligence were retained. One of the major mistakes artificial life inherits from artificial intelligence is a pervasive tendency to believe in the essential identity of a computer simulation and its physical realization. For example, in its strong form, biological computationalism comes very close to maintaining that computer simulations are "alive."

> *...living organisms are nothing more than complex biochemical machines*. However, they are different from the machines of our everyday experience. A living organism is *not* a single, complicated biochemical machine. Rather, it must be viewed as a large *population* of relatively simple machines. The complexity of its behavior is due to the highly nonlinear nature of the interactions between all the members of this polymorphic population. To animate machines, therefore, is not to 'bring' life to *a* machine; rather it is to organize a population of machines in such a way that their interactive dynamics is 'alive'. (Langton, 1989, pp.4-5)

I quote Chris Langton, not as an egregious example, but because he is among the most articulate and forthright advocates of this widely held position. There are a number of deep problems with this position.

First, a population of relatively simple machines interacting according to deterministic rules is a larger, more complex machine, regardless of the apparent complexity of its behavior.

Second, in Langton's sense of the term "linear system," automata are decomposable, linear systems *par excellence*. The input-output behaviors of program and simulation modules do not change with respect to computational contexts. Langton is using the adjective "non-linear" to mean "behavior of a whole which is not readily apparent without calculation" rather than "behavior of a whole which is not logically derivable from the behavior of its parts." There is a contradiction in his stated definition of linearity and his usage in the above passage (Langton, 1989, p. 41, also see app. 4 for extended discussion). In fact this is why computer simulations cannot be emergent at all, because computers *are* completely linear systems in the latter sense of the term.

Third, underlying this perspective on life is an extreme form of reductionistic platonism. Indicative is the following:

> Life is a property of *form*, not *matter*, a result of the organization of matter rather than something that inheres in the matter itself....It is effects, not things, upon which life is based -- life is a kind of behavior, not a kind of stuff. *Behaviors themselves* can constitute the fundamental parts of nonlinear systems -- *virtual parts*, which depend upon nonlinear interactions between physical parts for their very existence. Isolate the physical parts and the virtual parts cease to exist. It is the *virtual parts* that Artificial Life is after: the fundamental atoms and molecules of behavior. (Langton, 1989, p.41)

What is presumably meant here is that the objects of study are formal, platonic, *virtual* objects, objects of pure thought. Biological computationalism runs the risk of becoming a modern Pythagorean mysticism. Aristotle criticized the similar conceptions of nature held by their ancient counterparts:

> The 'Pythagoreans' treat of principles and elements stranger than those of the physical philosophers (the reason is that they got the principles from non-sensible things, for the objects of mathematics, except those of astronomy, are of the class of things without movement); yet their discussions and investigations are all about nature; for they generate the heavens and with regard to their parts and attributes and functions they observe the phenomena, and use up the principles and the causes in explaining these, which implies that they agree with the others, the physical philosophers, that the real is just all which is perceptible and contained by the so-called 'heavens'. But the causes and principles which they mention are, as we said, sufficient to act as steps even up to the higher realms of reality, and are more suited to these than to theories about nature. They do not tell us at all, however, how there can be movement if limited and unlimited and odd and even are the only things assumed, or how without movement and change there can be generation and destruction, or the bodies that move through the heavens can do what they do. (Aristotle, Metaphysics, I,8,990a, p. 705).

In holding an extreme platonic view, there is the obvious danger that the essential distinction between a simulation and a physical realization will be dissolved.

> There is a further epistemological danger in the belief that a high quality simulation can become a realization--that we can perfect our computer simulations of life to the point that they come alive. The problem, as we stated, is that there is a categorical difference between the concept of a realization that is a literal, substantial replacement, and the concept of simulation that is a metaphorical representation of specific structure or behavior, but that also requires specific differences which allow us to recognize it as "standing for" but not still realizing the system. In these terms, a simulation that becomes more and more "lifelike" does not at some degree of perfection become a realization of life. Simulations, in other words, are in the category of symbolic forms, not material substances. For example, in physics the simulation of trajectories by more and more accurate computation never results in realization of motion. We are not warmed by the simulation of thermal motions, or as Aristotle said, 'That which moves does not move by counting.' (Pattee, 1989, p.68)

In a similar vein, von Neumann also warned against the limitations of the axiomatic method, that in the idealization "one has thrown half of the problem out the window, and it may be the more important half" (von Neumann, 1949, p. 481; see app. 5 for an extended discussion).

Third, the computationalist view permits a blanket, *ad hoc* explanation for the existence of *all* order, whether produced by necessary or contingent relations, since it can be applied as an explanation for any observed behavior which exhibits order.

> There is a subtle, repeated process at work here. Order in a process is perceived and formulated as descriptive rules. From these, prescriptive rules are derived and imposed on a mechanical medium to allow simulation of the original process. The prescriptive rules are then projected back into the original process as cognitive agents, programs, accounting for the original order in terms of the simulated order. The working of the original is then said to be 'like' that of the imitation, and therefore due to the same kind of intentional control that created the imitation. To say it another way, order is abstracted from one system and imposed on a second, then the imposed order-as-program is abstracted from the second and projected onto the first. Rather than assuming that ontogenetic processes fit our notion

> of programs, we should be asking (and in fact people involved in computer simulation do ask) whether our notions of programs do justice to ontogenetic processes. (Oyama, 1985, p.62)

The ability to computationally generate any behavior once it has been defined gives rise to the illusion that this behavior must have been generated via similar means (see also Rosen, 1986, 1987). In universalizing computation to apply to all physical processes, this illusion is propagated and exacerbated. A reasonable antidote would be to admit non-computational processes into the picture, thereby distinguishing them from computations.

Biological computationalism needs to recognize its methodological limitations. One of these limitations is that computer simulations, being rate-independent digital processes, can represent but not realize rate-dependent analog processes such as measurement (Pattee, 1989). As Aristotle noted above, those motion-less relations between symbols cannot themselves implement motion. A similar fundamental limitation blocks the realization of the genotype-phenotype distinction in formal simulations.

Genotype and phenotype. Because the computationalist worldview does not admit of processes other than computations, all biological processes must be abstracted and redefined so that they fit into the computationalist explanatory frame. In this conceptual Procrusteanism, the differences between genotype and phenotype are truncated to exclude the all-important distinction between symbolic form and material substance.

> The most salient characteristic of living systems, from the behavior generating point of view, is the *genotype/phenotype* distinction. The distinction is essentially one between a specification of machinery -- the genotype -- and the behavior of the machinery -- the phenotype. (Langton, 1989, p.22)

Note that here both the specification and the behavior are in the "machinery," which to Langton means the symbolic machinery (see also Langton, 1984, p.136). Both the specification and the behavior are conceived in purely symbolic terms. In its reduction to a purely formal distinction, the symbol-matter relation has been eliminated completely. What is traditionally understood as a semantic, symbol-matter relation is transformed into a completely syntactic one.

Contrast this with another simple analogy of the genotype-phenotype distinction, which *does* preserve the symbol-matter relation, that of the blueprint of a house and the house itself (Oyama, 1985, p. 46). The blueprint of the house is a symbolic configuration which directs the complex material fabrication by human construction workers of a material, nonsymbolic object, the house.

In the computationalist conception, on the other hand, the genotype-phenotype distinction is a distinction between strings labelled "genotype" from those labelled "phenotype," completely on the basis of an arbitrary choice of what notation one uses to describe the machine. The supposedly essential distinction between the specification of the state-transition rules of a computational device and its consequent behavior collapses under close scrutiny. This quite innocent-looking distinction, which also forms the basis for an autonomous "symbol level" in the theories of Newell (1980) and Pylyshyn (1984), is discussed at length in chapter 7 and in appendices 2 and 4. The upshot of all of this is that as long as we are talking about a Turing machine with a finite tape, the specification of the machine and its behavior can always be combined into a global state description (cf. Minsky, 1968, p.24) of the machine and its behavior. In this notation, which is a formally equivalent relabelling of the specification/behavior notation, the purportedly essential distinction between specification and behavior, and hence the simulated genotype and phenotype, dissolves into an arbitrary labelling of states. This is also a prime example of how potential infinities are used ideologically to essentialize central theoretical distinctions of a theory (app. 1), thereby insulating them from criticism, for if potentially infinite tapes *are* allowed, then the collapsing of the specification/behavior distinction into the global state description is no longer possible in the general case. To cross the "trip-wire" of criticizing this distinction immediately immerses one in a contentious, fundamental philosophical debate about the foundational status of potential

infinities in mathematics and the nature of mathematical entities themselves. The deterrence effect of this imposing edifice is readily apparent.

The criticisms levelled here are expressly *not* directed at all artificial life simulations of morphogenesis, only at the ontological assumption that morphogenesis consists solely of computations. There is no question that simulations of ontogenetic processes as "developmental programs" are useful in terms of exploring what kinds of generative strategies are available (Lindenmeyer systems and chaotic generators immediately come to mind), but we must ask ourselves how computationalist ideology affects our visualization of the process and if it ultimately narrows the scope of thought to one class of mechanisms.

Elimination of the semantic and the pragmatic. It could be argued, for example, that computationalism, in its totalizing strong form, potentially impoverishes biology's conceptual basis by obliterating alternative, non-computational forms of explanation. When computations become the only processes imaginable, then it becomes impossible to think in terms of other functional considerations of semantic and pragmatic character. Rosen(1986) argues that this has already happened with the Newtonian paradigm, of which computationalism is an extreme variant. He demonstrates that the basic explanatory structure of the Newtonian paradigm is incompatible with Aristotle's notion of final cause, which here are embodied in pragmatic relations. We have seen above how the semantic genotype-phenotype relation has been eliminated by computational syntactization. If we adopt the Newtonian paradigm wholesale, without criticism, reservation, or alternative, we stand to lose the whole constellation of concepts tied up with utility and purpose, a situation which, where it has happened, "has been disastrous for biology" (Rosen, 1986, p. 109).

Towards a cognitive biology

We have seen that none of the preceding accounts incorporates all of the important aspects of symbolic organization and function: syntax, semantics, and pragmatics. The pragmatic aspect is quite important to us from the standpoint of using the functional organization of biological symbols to design devices that have useful functions.

Within theoretical biology, there have been attempts to develop a "cognitive biology" or "cell psychology" (Pattee, 1982, 1985; Kelly, 1988; Goodwin, 1978, 1982; Oyama, 1985; Rosen, 1974, 1985; Boden, 1981; Piaget, 1971, 1976; Minch, 1988) in which the essential, most rudimentary cognitive relations of primitive organisms are uncovered, explicated, abstracted, and then utilized as design principles.

> Where can the cognitive sciences expect to find more promising foundations for symbol-matter theory? I see no course but to look at the first 'rungs of the ladder.' We need to look for simple embodiments of *natural* matter-symbol systems with both empirical power and conceptual generality. Why should we only work with the ultimate functional complexity of brains, or the ultimate artificiality of computers, or the ultimate meanings of philosophical discourse? As a trial first rung I suggest trying to adapt our fundamental concepts of cognitive science to the basic symbol-matter problems of biology, and even physics, where a few rungs are already secured. (Pattee, 1982)

The design principles we shall discuss are similar to the tripartite, semiotic analysis of DNA outlined in chapter 2. Recall that the syntactics of DNA are found in the transcription-translation coding, the semantics are the effects of the produced, folded protein on the cellular environment and beyond, and the pragmatics have to do with the survival value of a particular DNA sequence and its phenotypic product. This set of relations, of course, plays itself out over successive generations of the organism.

Perception, cognition, and action. It is simpler, especially with the design of learning devices in mind, to consider the ongoing relations of cellular symbols and the cellular environment within a single generation. This will consist of the effects of the environment on the internal "biological state" of the cell, the "biological state" transitions effected in response to external stimuli, and the resulting action

produced by the cell. Here the "biological state" loosely refers to the set of bio-molecules being currently expressed and their relations. An example of changes in "biological state" would be cascade of internal molecular changes (changes in secondary messengers and gene expression) which occurs following the binding of a hormone to a cell's membrane-bound external receptor molecules. The pragmatics here would involve whether the cell by its responses has improved the cell's situation with respect to its structural stability and persistence. We have, on a cellular level, the rudiments of perception, coordination, and action.

> In speaking of "perception" and "cognition" we are not referring to concepts extraneous to the physical world, implying the mind of a mysterious "inner man." Rather, we realize that the beginnings of perception are already manifest in microorganisms, as represented by such phenomena as phototaxis in bacteria and phototropism in fungi. Even at this primitive level, the organism receives signals from the outside world, evaluates their significance, and responds appropriately. What do I mean when I say a bacterium or a fungus "evaluates" the significance of signals? I mean no more, and no less, than that the organism possesses structures that permit him to react appropriately, that is, in an evolutionarily adaptive manner. What do I imply by referring to a bacterium or a fungus as "him"? Do I refer to a subject with a consciousness, with a mind? No, I simply refer to the organism as an indivisible object, a functional unit, and, as such, an individual. (Delbrück, 1986, p.273)

The interaction of an organism with its environment can be considered as an ongoing sequence of perceptions of the environment and actions on it. Alternatively, we can view this loop as actions on the environment and perception of the consequences. Mediating between perceptions and actions are coordinations, which allow for flexible perception-action combinations.

Each process has a corresponding biological structure. Perception is carried out by sense organs, which can be simple molecular receptors making one distinction on the world or complex organs making many at once. Coordination is carried out by a coordinative system, generally a hormonal system or the nervous system. Action is carried out by diverse effectors, which can be motor organs, appendages, secretory organs, etc.

Sensors. Sensors are the physical structures which produce perceptions, encoding nonsymbolic, interactions with the environment into discrete symbols which can be manipulated by coordinative structures. Sensors allow the organism to make distinctions upon the world, to react contingent upon distinguishable states of the world.

To the extent that discrete distinctions are being made, as in the classic case of the various specialized receptors in the eye of the frog (Lettvin *et al*, 1959), we can say that a symbol-producing process of measurement is taking place, linking the world to particular discrete states of the organism. In such cases it is possible to speak of the semantics of such states as the relationship implemented between the organismic states and the world at large by the sense organs.

Coordinators. Coordinative structures map the distinctions made by sensors to actions that are to be taken by the device or organism. "... the function of the mind, in biological terms, is to act as a transducer between percepts and specific actions, mediated by the organism's effector mechanisms" (Rosen, 1985, p. 46). These coordinative structures can be very complex, involving memory, logical operations, and pre-existing processing structures. They can involve all of the possibilities envisioned by the computational cognitive psychologists with their "information processing" models and all of the highly developed algorithms of contemporary artificial intelligence. In biological systems, most fast coordinative function is carried out by the nervous system, while slower, more generalized coordination tends to be achieved by hormonal diffusion through various types of circulatory systems (cardiovascular transport, cerebrospinal fluid, etc).

To what degree can we say that the coordinative system of an organism is syntactic in character? While we do not want to forget that the nervous system has analog, stochastic and chaotic components, and is therefore seldom deterministic in the sense that we know what the rules are, we are justified in calling coordination a syntactic function to the extent that there are reliable, identifiable mappings

between discrete perceptual inputs and discrete behavioral alternatives.[1] How much of coordination involves syntactic mechanisms depends upon the organism being examined, how well the coordinative mechanisms are understood, and to what degree those coordinations are reliable enough to be adequately described in terms of rules. An organism with more regular, predictable behavior will appear to have a more syntactically-describable coordinative part--its input-output function will be more amenable to description by symbolic rules. On the other hand, an organism with more stochastic (or "sloppier") percept-action coordination will appear to have less in the way of behavior describable through syntactic relations.

Effectors. Effectors are the physical structures which produce actions given input symbols from the coordinative part. Examples of effectors are the wings of the bird, the mechanical arm of the robot, the jaws of the alligator, the spinerettes of the spider, the teeth of the beaver. To the extent that the effectors are activated by discrete switchings to produce continuous motion or action, we can say that a semantic relation is being implemented, connecting the discrete, rate-independent symbolic stimulus of the effector organ to some continuous, rate-dependent nonsymbolic effect on the environment.

Organism-environment relations. The simplest framework involving these functionalities is the one illustrated in figure 3.1. At any given time the organism is interacting with the environment, making distinctions on it, deciding what actions to pursue, and changing the environment through those actions. Thus, the perception-cognition-action loop constitutes a constant set of organism-environment inter-relationships.

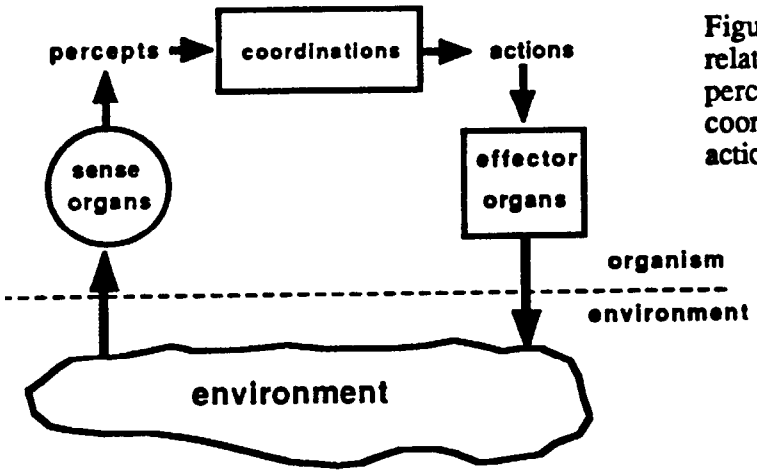

Figure 3.1 Structure-function relations between sensors and percepts, coordinators and coordinations, effectors and actions

This framework need not apply to the entire organism; it does not necessarily imply centralized control. A given organism might in fact have many such loops going into and out of the environment, and these loops might act semi-independently of each other. A prime example of this is the existence of ganglionic reflex arcs in vertebrates. Thus multiple perception-coordination-action systems in parallel might be necessary to replicate the behavior of real organisms or to design effective robots (Brooks, 1986, 1987).

This schema is still static, however. While there is a feedback loop running from the environment to the organism's perceptual apparatus and back to the environment by means of the organism's coordinative structures and its effectors, there is no means by which the organism's perceptual,

[1] There is, of course, a large literature in cognitive science and computational psychology which studies cognitive systems *exclusively* in syntactic terms (e.g. Pylyshyn, 1984; Lachman *et al*, 1979). There is also a smaller ecological realist literature disputing this analysis in favor of analog mechanisms (e.g. Kugler *et al*, 1980; Carello *et al*, 1984). This controversy will probably continue until the neural mechanisms are understood in more detail.

cognitive, and behavioral structures can be altered by experience. As represented thus far, the organism cannot learn or evolve.

The evolutionary advantages of cognition. Even a static set of perceptual, cognitive, and behavioral structures is better than operating blindly in an environment, regardless of any external conditions. Above all, these operations enable an organism to make its actions contingent upon the perceived state of the external environment, thereby increasing the probability of its subsequent survival.

> To the extent that present demands resemble those of the past, the strategy of evolving according to patterns that were successful in the past is sound. It is better than responding without direction to every random gust of selection pressure. However, a species does even better to direct its evolution in accordance with the current state of the environment and the organism than the past. For sensory evolution, an organism detects relevant information about its environs and influences the way it evolves accordingly. This capacity might be dismissed as an impossible ability for a species to evolve had we not found mechanisms for it. (Campbell, 1985, p.146)

The advantages of increasing perceptual and behavioral repertoires as well as increasing the flexibility of the cognitive apparatus are relatively obvious. If two birds have similar behavioral repertoires but one has color vision and the other does not, the bird with color vision will be able to discriminate situations that the second animal cannot. This can translate into a survival advantage, if poisonous insects are one color while edible ones are another. If two species of lizard have similar perceptual repertoires as well as nearly identical behavioral ones, but one has the option of shedding its tail in critical moments, that animal may survive where a non-tail-shedding species may not. Similarly, those animals with more flexible percept-action schemata will be able to profit better from experience than their evolutionarily "hard-wired" counterparts.

The disadvantages of increasing perceptual and behavioral repertoires arise in the increased complexity of the more complex organs, and their higher overhead in terms of energy, material, and development time.

Variation, selection, and propagation. Evolution and learning in their most general aspects can be seen as variation-selection-propagation-variation cycles (figure 3.2). In biological organisms, variations are produced by genetic mutations or thermodynamic fluctuations, which form alternative

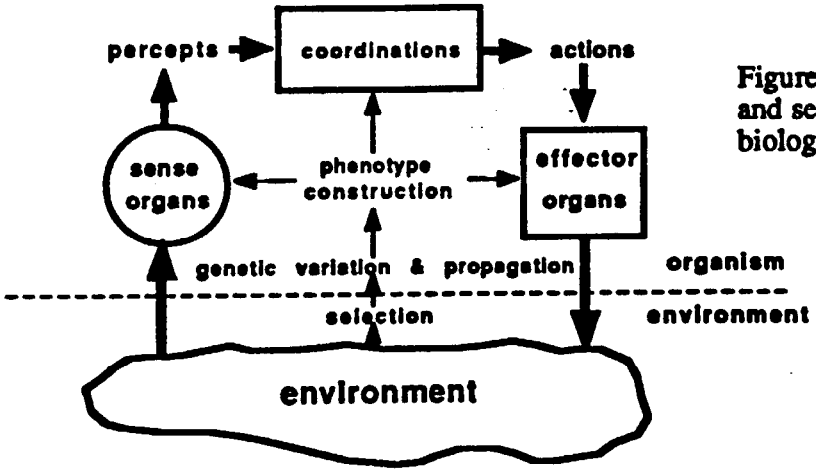

Figure 3.2 Structural plasticity and selection relations for biological organisms

phenotypes, which undergo selection, the result of which is a new set of genes, which then undergo mutations. Interactions of the organism with its environment thus come to determine, along with internal, organizational, constructional constraints (e.g., "Galton's polyhedron" in Gould, 1981; Riedl, 1978), the structures which carry out the functions of perception, coordination, and action. The general process by which the symbolic genotype gives rise to a nonsymbolic phenotype will be called *construction.*

As we shall see in the discussion of syntactically-adaptive and semantically adaptive devices (chaps. 9 & 10), this selective feedback loop has its analogues in artificially constructed learning devices. For these trainable machines, functional variations are achieved through optimization programs in the "software" part and instructions for fabricating and calibrating the sensors and effectors in the "hardware" part. von Neumann's description of a self-reproducing kinematic automaton comes the closest to making all of these constructional and instructional processes explicit. Selection is achieved by some measurement of the performance of the device and the consequent alteration of its internal structure. Inheritance is achieved by memory in the programs and in the instructions for constructing the hardware. Memory allows comparison between performance of the present structure with the last one and the propagation of the "better" one. In this case what is "better" performance is determined by the user's goals.

Time scales. With the introduction of this last "structural feedback" loop we have two time scales, that of the perception-coordination-action cycle and that of the construction-performance-selection cycle. In biological systems, these time scales are quite far apart: with behavioral responses on the order of seconds and structural responses on the order of millions of years. If we also include an ontogenetic feedback loop, learning, which can be seen as a form of somatic selection (e.g. Edelman, 1987), then we have a third time scale on the order of an individual's lifespan.

Possible arenas of adaptation. In the perception-coordination-action schema, each function can be an arena for adaptation through variation and selection. Contemporary conceptions of learning devices almost completely focus on coordinative, "cognitive" adaptation, neglecting the possibility for adaptation in the functionalities of perception and action. But the functionalities of perception and action are the ones which define the relations of the cognitive categories to the world at large. Once we also have the means of altering semantic relations adaptively, we achieve what Pattee has called *semantic closure*, a prerequisite for full epistemic autonomy (Pattee, 1985). Arguably, once we have all the primitive functionalities integrated in a semantically closed way we have the preconditions for the formation of an autonomous evolutionary "subject" (Beurton , 1981).

> I will start from the idea, stressed by ethologists such as Thorpe, that the behavior of animals, like that of computers, is programmed; but that unlike computers, animals are self-programmed. The fundamental genetic self-program is, we may assume, laid down in the coded DNA tape. There are also acquired programs, programs due to nurture; but what can be acquired and what cannot--the repertoire of possible acquisitions--is itself laid down in the form of the fundamental genetic self-program, which may even determine the probability or propensity of making an acquisition. (Popper, 1987, p. 151)

Leaving out adaptation of these semantic functionalities forecloses the possibility for the organism or device to construct its own cognitive relations to the world; hence, the possibility for new cognitive primitives, for the device to have cognitive autonomy, is disallowed.

An integrated framework. In this scheme we have integrated syntactic relations (coordinations), semantic relations (perceptions, actions), and pragmatic relations (selection) into a unified evolutionary framework. Because of its high level of abstraction and generality of the relations, many existing psychological paradigms can be seen in these terms. The scheme can incorporate essential features from both Darwinian selection and thermodynamic bifurcation within the concept of construction processes. The categories of analog-to-digital, digital-to-digital, and digital-to-analog processes, which von Neumann employed in his schematic of the brain (von Neumann, 1958), correspond to perception, coordination, and action. His digital genotype and analog phenotype have their equivalents in the symbolic to nonsymbolic process of construction. Computational psychology, with its input and output transducers, internal symbolic representations, and "information processing" functionalities (e.g. Lachman & Lachman, 1979; Pylyshyn, 1984) can also be fit into the basic categories, although not without a substantial shift in emphasis away from the predominant role of syntactic coordinations.

The important, ecological realist, direct perceptionist, Gibsonian account of the embedding of the organism in its environment (e.g. Gibson, 1966; Michaels & Carello, 1981) is more difficult to incorporate because this tradition has generally rejected symbols and representations as coordinative structures. The strength of this tradition has been in its emphasis on perception and action and the evolution of these functionalities. It draws some of its theoretical underpinnings from dynamical theory and nonequilibrium thermodynamics (e.g. Kugler *et al.*, 1980; Kelso *et al*, 1988, Shöner & Kelso, 1988). Perhaps the perspective might be at least partially accommodated if the symbols of the coordinative structures could be seen as discrete, dissipative attractor basins and the coordinations themselves as switchings between these basins. The full development of such a theory would constitute an important advance in understanding the emergence of discrete symbolic structures from the dynamical flux. Related to this perspective is their observation made that self-complexification cannot happen in a purely digital system, that emergent structures and functions can occur only if analog processes are also present (Carello *et al.*, 1984; also quoted in chapter 7 here).

The basic symbolic relations of various traditional psychologies, going back to William James and all the way back to Aristotle can also be found here. Aristotle's four types of explanation (or "causes") can also be found: formal cause in syntactic coordinations; material cause in the physical structures of the sensors, effectors, and coordinators; efficient cause in both the sensory inputs from the environment and the actions the organism performs on the environment, and final cause in the pragmatic loop which relates stability and survival with the ongoing (re)construction of the organism (cf. Rosen, 1986).

Most important for our purposes, it affords a rich framework for analyzing the functional characteristics of various devices around us, and provides the basic plan for constructing new ones with different adaptivities. This framework will serve as the basis of the taxonomy of adaptive devices developed and analyzed in chapters 6 through 11. The elements of the framework will be refined and their definitions made more precise in chapter 5.

As we shall see in the next chapter, this framework is also very similar to that of the modelling relations of physics. Out of the historical explication of modelling relations and the experience of quantum mechanics, came the realization that the various syntactic, semantic, and pragmatic relations involved in the act of modelling physical systems are irreducible epistemic functionalities. By another route we are again led back to the concept of semantic closure.

Chapter 4 The physics of symbols

> When you can measure what you are speaking of and express it in numbers you know that on which you are discoursing. But if you cannot measure it and express it in numbers, your knowledge is of a very meagre and unsatisfactory kind. (Lord Kelvin's dictum, in Porter, 1966, p.34).

> Mathematicians and physicists were first to gain clear awareness of this symbolic character of their basic implements.2 The new ideal of knowledge, to which this whole development points, was brilliantly formulated by Heinrich Hertz in the introduction to Principles of Mechanics. He declares that the most pressing and important function of our natural science is to be able to foresee future experience-and he goes on to describe the method by which science derives the future from the past: We make "inner fictions or symbols" of outward objects, and these symbols are so constituted that the necessary consequences of the images are always images of the necessary consequences of the imaged objects. (Cassirer, 1955, p. 75; 2. This is discussed in greater detail in my book *Zur Einstein'schen Relativätstheorie* (Berlin, B. Cassirer, 1921); cf. especially the first section on "Massbegriffe und Denkbegriffe.")

Physics was the first science to construct precise, rigorous *formal* theories of the world, theories relating the operation of rules upon symbols to the law-like behavior of matter. While the pre-Socratic Greeks had notions that thought itself might be symbolic in character, something like their mathematical forms, it remained for Aristotle and subsequent physicists to relate these symbols more and more explicitly to the external world and to successively clarify the nature of this relation.

Aristotle is generally held to be the first to consider specific observable factors which determine motion. In his Physics he proposed that the two main factors which determine an object's speed are its weight and the density of the medium through which it travels. More importantly, he recognized that there could be mathematical *rules* which could describe the relation between an object's weight, the medium's density and the consequent rate of fall:

> (1) in natural motion-that is, in the case of freely falling or freely rising bodies-the speed is proportional to the density of the medium. And (2) in forced motion, the speed is directly proportional to the force applied and inversely proportional to the mass of the body moved... (Lloyd,1968, p.176).

This was the first time that observable quantities had been expressed in symbolic, numerical form, allowing the results of observations to be used in calculations.

While these quantities were expressed in terms of numbers, they were still generally regarded as inherent properties of the objects themselves. It was not until Galileo took the interrelationships of the signs themselves as the objects of study that we even see the beginnings of what was to be progressive dissociation of the symbols from the objects represented. Galileo's insight was that the symbols themselves and their interrelations could be studied mathematically quite apart from the relations in the objects that they represented. This process of abstraction was further extended by Newton, who saw that symbols arising from observation, thereby representing initial conditions, are distinct from those involved in representing the physical laws which govern the subsequent motion.

The successive explication of the modelling relation

Following Newton, the Kantians and the British empiricists paved the way for a theory of how symbols could be related to the world in physical models by constructing a plausible account of how we could have knowledge through the association of sensory experiences. Not until the 19th century, however, was the modelling relation itself dealt with in a highly self-conscious way. While the Newtonian paradigm was having unparalleled successes in explaining an enormous range of disparate

phenomena, there was little motivation in probing the grounds for its "unreasonable effectiveness." Perhaps, partly due to the inability of Newtonian mechanics at the time to quickly incorporate the rising, observationally-driven fields of thermodynamics and biological evolution, the epistemological assumptions of the Newtonian worldview began to be examined more closely. In our century a similar wave of questioning has been generated by the discovery of quantum mechanical properties, culminating in the pragmatically-oriented, operationalist methodology of Neils Bohr. During such periods, it is natural to re-evaluate and clarify one's methodological framework in the hopes of firmly grounding one's knowledge in what is most unambiguous and least subject to interpretational dispute--in the symbols themselves.

The physiologist and physicist Hermann von Helmholtz was one of the first scientists to develop a semiotics of science, combining insights from his study of perception with those from experimental physics. Von Helmholtz saw that perception and measurement played analogous roles in organisms and physical theories and that both were bound up with the production and use of symbols. He further recognized that a symbolic form produced by perception or measurement need not necessarily be like the object being perceived or measured to be useful, as long as its dependence upon that object is stable:

> Our sensations are only effects which are produced by external agents upon our organs, and the way in which such an effect is manifested, of course depends essentially upon the nature of the apparatus which is affected. As far as the quality of of our perception gives us information about the characteristic nature of the external source of the stimulation, it can be regarded as a symbol, but not as an image of this source. For an image some kind of identity with the portrayed object is demanded: for a statue, identity of form; for a drawing, identity of perspective projection in the field of vision; and for a picture, the additional identity of colors. A symbol however does not have to possess any kind of similarity with the object it represents: the relation between them is confined to the fact that the same object, acting under the same circumstances, will produce the same symbol; and that unlike symbols thus always correspond to unlike influences. (von Helmoltz, 1878, p.212)

It is this stability of perception and measurement coupled with the ability to compare symbolic identities that make the representation of natural law possible at all:

> Contrary to popular opinion, which accepts in good faith the complete truth of the images which our senses furnish us of external things, the morsel of similarity which we have acknowledged may appear quite insignificant. Actually it is not, for by it something of the greatest importance can be achieved, namely the representation of lawfulness in the processes of the real world. Every natural law states, that preliminary conditions which are equivalent in a certain respect, must have consequences which are equivalent in other respects. Since in our perceptual world equivalence is indicated by the identity of symbols, the law of nature that the same consequences follow from the same causes will have a corresponding law concerning the consequences which are just as regular in the field of our perceptions.
>
> If ripening berries of a certain kind simultaneously develop red pigment and sugar, in our perception red color and sweet taste will always be found together in berries of this kind.
>
> Thus, even if our sensory perceptions in their quality are only symbols, whose special nature depends entirely on our organization, they must not be discarded as empty appearances; rather they are symbols for something, either existing or happening--and what is most important--they can represent the laws governing this event for us. (von Helmoltz, 1878, pp.212-213)

Perhaps most importantly, von Helmholtz saw these symbolic results of perceptions and measurements "not as empty appearances," but having value in relation to the purposes of the observer.

The Hertzian paradigm

In 1894 Heinrich Hertz published his *Principles of Mechanics* which attempted, like Mach's earlier *Mechanics*, to purge mechanics of metaphysical, mystical, undefined, unmeasured entities such as force and to base the theory explicitly on measurable quantities. Hertz wanted to be as clear, rigorous, and concise as possible, so that implicit, and perhaps unnecessary, concepts could be eliminated from physical theories. If one begins to deny the validity of talking about physical properties which one

cannot measure and only infer, and the only knowledge of the observable properties is through measurement, then one is left with only with the results of the measurements, which are symbols. Physical theory becomes about the correspondences of observationally-derived symbols in *models*, what Hertz called "images." In his images he distinguished a logical, syntactic part, an empirical semantic part, and an intentional, pragmatic part:

> It is not going too far to say that this representation [the customary (Newtonian) representation of mechanics] has never attained scientific completeness; it still fails to distinguish thoroughly and sharply between the elements in the image which arise from the necessities of thought, from experience, and from arbitrary choice. (Hertz, 1894, p.8)

The syntactic lies in the logical "necessities of thought," the semantic "from experience," and the pragmatically driven "arbitrary choice" of what to study. In the very beginning of his Introduction to *The Principles of Mechanics,* Hertz begins with the pragmatic aspects of models, "the anticipation of future events" and proceeds to relate these to both their syntactic aspects, in the "necessary consequents of the images in thought," and their semantic ones, in "the images of the necessary consequents in nature of the things pictured":

> The most direct and in a sense the most important, problem which our conscious knowledge of nature should enable us to solve is the anticipation of future events, so that we may arrange our present affairs in accordance with such anticipation. As a basis for the solution of this problem we always make use of our knowledge of events which have already occurred, obtained by chance observation or by prearranged experiment. In endeavoring thus to draw inferences as to the future from the past, we always adopt the following process. We form for ourselves images or symbols of external objects; and the form which we give them is such that it is such that the necessary consequents of the images in thought are always the images of the necessary consequents in nature of the things pictured. In order that this requirement may be satisfied, there must be a certain conformity between nature and our thought. Experience teaches us that the requirement can be satisfied, and hence that such a conformity does in fact exist. When from our accumulated experiences we have once we have succeeded in deducing images of the desired nature, we can then in a short time develop by means of them, as by means of models, the consequences in which the external world only arise in a comparatively long time, or as a result of our own interposition. We are thus enabled to be in advance of the facts, and to decide as to present affairs in accordance with the insight so obtained. The images which we here speak are of our conceptions of things. With the things themselves they are in conformity in *one* important respect, namely, in satisfying the above mentioned requirement. For our purpose it is not necessary that they should be in conformity with the things in any other respect whatever. As a matter of fact, we do not know, nor do we have any means of knowing, whether our conceptions of things are in conformity with them in any other than the *one* fundamental respect. (Hertz, 1894, pp. 1-2)

Thus Hertz in one concise formulation relates syntactic, semantic, and pragmatic functionalities. We will go through this set of functional interrelationships step by step. We will start with the relations between the syntactic part and the semantic parts. The kind of relationship Hertz had in mind is depicted in figure 4.1.

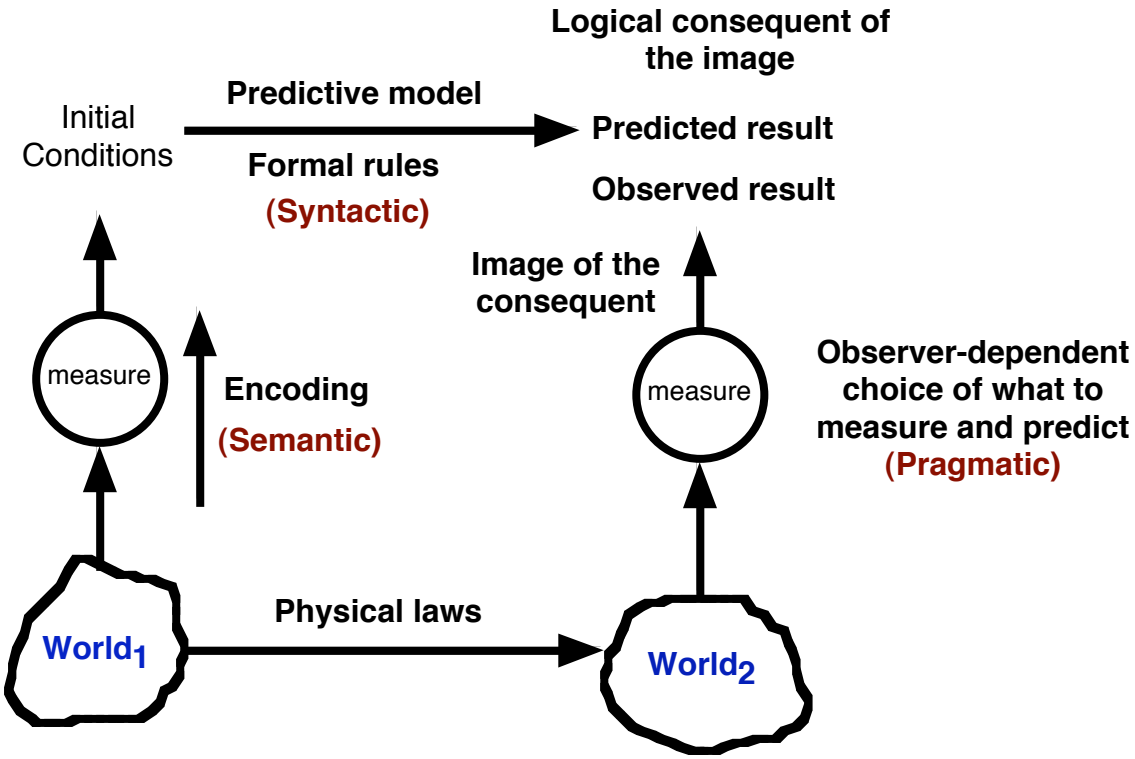

Figure 4.1 The Hertzian modelling paradigm.

The "inner fictions or symbols" are the results of measurements, the outward objects are otherwise not delineated, "the necessary consequences of the images" are the logico-mathematical consequences of measurements in the physical theory, and the "images of the necessary consequences of the imaged objects" are the measured changes brought about by the action of natural laws. Hertz says that the natural laws and the imaged objects can never be known directly; it is only through this indirect logico-empirical functional relation that knowledge of them can be asserted. This shift decisively places the problem of explaining the action of symbols at the heart of epistemology:

> Heinrich Hertz is the first modern scientist to have effected a decisive turn from the copy theory of physical knowledge to a purely symbolic theory. The basic concepts of natural science no longer appear as mere copies and reproductions of immediate material data; rather they are represented as constructive projects of physical thinking--and the only condition of their theoretical validity and significance is that their logical consequences must always accord with the observable data. In this sense, the whole world of physical concepts may now be defined as a world of pure signs, as was done by Helmholtz in his theory of knowledge. If we compare this formulation with the epistemological presuppositions of the "classical" theory of nature, a strange contrast becomes evident. In interpreting sensuous qualities as mere signs (puri nomi), Galileo severed them from the objective world of natural science, They now bore the character of the conventional, accidental, and arbitrary, in contradiction to the objective necessity of nature, Knowledge must overcome and cast off everything that is merely significative, in order to penetrate to authentic reality. But now the line dividing subjective appearance from objective reality is drawn in a new sense. Both have a purely indicative character; they are merely indices of reality--and the only difference between them is that their indications possess a different value, a different theoretical significance and mode of universality. Thus the concept of symbol has become a center of the whole epistemology of physics. (Cassirer, 1957, pp.20-21)

Robert Rosen has further explored and deconstructed some of the the Newtonian assumptions that both underlie and are transcended by Hertz' scheme, in the process extending the Hertzian picture of the modelling relation. The assumptions of a law-governed material order and a logical, rule-governed mental order form the basis of the Newtonian conception of natural law:

> First is the belief that the sequence of events that we perceive in the external world is not entirely whimsical, chaotic, and arbitrary but obeys definite laws or relations. The relations that exist between events in the external world, and that govern their succession, collectively constitute what we call causality.
>
> ...The other part is an independent belief that this causal order relating events can be, at least in part, grasped and articulated by the mind. It is a belief that, in some deep sense, the causal order relating events can be mirrored in a corresponding relation between propositions that describe these events. Now such propositions belong to an internal, symbolic, linguistic world and hence cannot themselves be related by any kind of 'causality.' But there is another kind of order through which propositions can be related, and that is a logical order or implication. (Rosen, 1985, p.179)

Here causality is a law-like, universal, necessary relation, where the other kind of order is a rule-like, locally applied, convention. Both appear to be determinate orders, but one is unchanging and the other is an arbitrary convention selected for its correspondences with observed behavior.

> Thus we must also believe that this causal order relating events in the external world, can be brought into congruence with a logical or implicative order in some appropriate logical, symbolic world of propositions describing these events. When such a congruence is established, *implications* in the logical system become *predictions* about the causal order. (Rosen, 1985, p.179)

Rosen summarizes this relation in a diagram (figure 4.2), relating symbols to the world at large. The diagram is more general than Hertz's in that it encompasses both encoding (in Hertz's terms, measurement) and decoding processes.

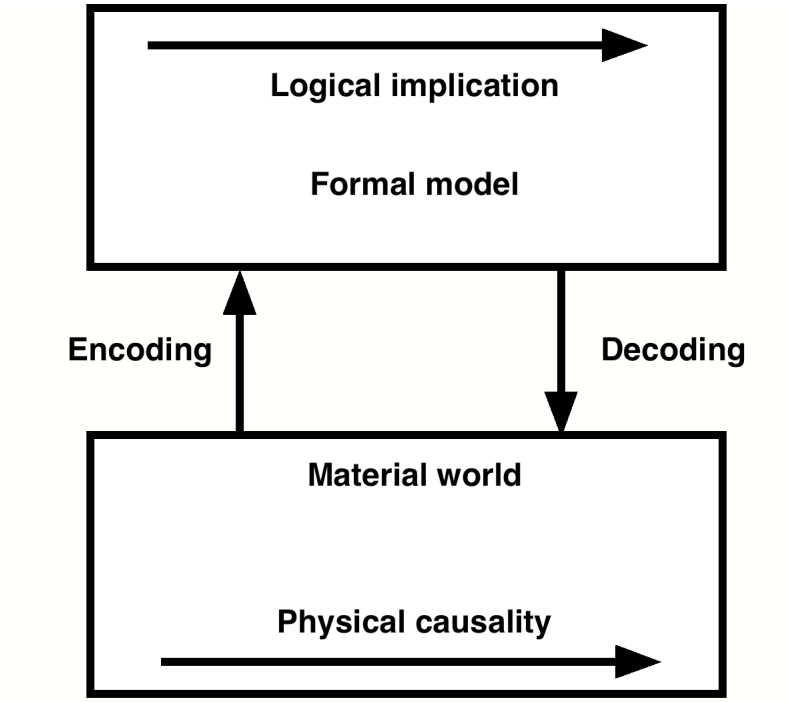

Figure 4.2 Rosen's extended set of basic modelling relations.

The underlying assumption of Newtonian mechanics is that systems can be broken down into structureless particles, having only their constitutive, time-independent parameters (e.g. mass) as well as non-constitutive, time-dependent ones (e.g. position). The task is to derive the non-constitutive parameters from the constitutive ones for any point in time, given the initial values for the non-constitutive ones. The "sweeping implications" of this strategy are:

(1) Insofar as *any* material system can be resolved into a system of structureless particles, Newtonian mechanics seems to provide a recipe, or algorithm, for the study, modelling, and representation of *any* system. That is, it in principle enables us to construct the diagram of fig. [4.2], for *any* material system on the left-hand side. It establishes simultaneously the nature of the formal, mathematical image *and* the encoding and decoding arrows that convert the formal system into a model.
(2) Once the Newtonian picture is accepted, it becomes a purely *empirical* problem to determine, for a given system of interest, what are its constituent particles and what are the forces imposed upon them....
(3) The encoding and decoding arrows in fig. [4.2] that Newton posited have become axiomatic, and thus in effect invisible. They are no longer recognized as the pivots on which the whole picture turns, but have become as necessary a part of scientific thought as Euclidean geometry was prior to 1800. (Rosen, 1985, pp.182-183)

The encoding and decoding relations are crucial to the semantics of symbols utilized by physical theories, biological organisms, and constructible devices, but they are virtually ignored in the description of scientific theories and in the field of artificial intelligence. Very few physics textbooks today have anything like a Hertzian commutation diagram and those that do may not explicitly state that the "images of the consequents of nature" are the symbolic outputs of the action of material measuring devices. And even when the measurement process is explicitly taken into account, the physical construction, calibration, and preparing of the experimental apparatus is generally not included in the picture. These nonsyntactic, material aspects of scientific theories, "the pivots on which the whole picture turns," have been made invisible. This may be due to latent platonic and realist assumptions which have all along dominated classical physics, mathematics, and the philosophy of science. In our century it is tied up with the general syntactization of meaning, of the reduction of observations to logical statements which were discussed in chapter 2.

The belief that all natural systems can be encoded into bundles of independent, invariant properties leads to "the fundamental reductionist assumption that, among all possible encodings of a natural system, there is a *biggest* one, which maps *effectively* on all the others." (Rosen, 1985, p.184) "This 'largest' description manifests every shred of physical reality of every system it images; it is a full syntactization; a transcription, not an abstraction." (Rosen, 1987, p.13) In short, the Newtonian paradigm supports the notion of an "objective" reality of finite depth, whose full image of which every other partial image must be a part.

Modelling relations and complementarity principles

It was just this unshakable belief in one underlying "largest" description which twentieth century discoveries shook to the root. Classical physics had assumed that the world operated independently of any measurements on it, and that therefore, the measuring process could be ignored in the description of a scientific theory. Since the act of measuring did not change the physical system being measured, it was possible to construct "true" representations of the physical system independent of the particular measurements. Many of the phenomena encountered by the early physics lent themselves to this "objective" treatment and interpretation.

In the twentieth century, however, physics began to encounter phenomena which defied an kind of unified classical description, for the first time making it completely clear that the representational account of observation had to be abandoned. As a consequence, measurement itself would have to be taken as a primitive process and no scientific model would be complete without an account of how to build and calibrate the measuring apparatus. Niels Bohr, champion of this view, demanded that the process of observation be included explicitly in the description of scientific practice:

On the lines of objective description, it is indeed more appropriate to use the word phenomenon to refer only to observations obtained under circumstances whose description includes an account of the whole experimental arrangement. In such terminology, the observational problem in quantum mechanics is deprived of any special intricacy and we are, moreover, directly reminded that every atomic phenomenon is closed in the sense that its

observation is based on registrations obtained by means of suitable amplification devices with irreversible functioning, such as, for example, permanent marks on a photographic plate, caused by the penetration of electrons into the emulsion. In this connection, it is important to realize that the quantum mechanical formalism permits well-defined applications referring only to such closed phenomena. Also in this respect it represents a rational generalization of classical physics in which every stage of the course of events is described by measurable quantities. (Bohr, 1954, p.73)

The process of measurement irreversibly creates macroscopically observable symbols, which constitute the phenomenal domain, and whose relevant properties, those of "type" can be unambiguously agreed upon and communicated. A set of measuring devices and experimental arrangements creates an *observational frame* and, consequently, a stable phenomenal domain. Once fixed by the observational frame, the domain is 'closed' because of the closed set of possible measurement outcomes needed for unambiguous replication and communication of results. Open possibilities for the results of measurements would lead to ill-defined phenomena. The result of this is that a explicate, well-defined, symbolic, "classical" domain is created out of an implicate, ill-defined, nonsymbolic domain:

...however far the phenomena transcend the scope of classical physical explanation, the account of all evidence must be expressed in classical terms. The argument is simply that by the word 'experiment' we refer to a situation where we can tell others what we have done and what we have learned and that, therefore, the account of the experimental arrangement and of the results of the observations must be expressed in unambiguous language with suitable application of the terminology of classical physics.

This crucial point...implies the *impossibility of any sharp separation between the behavior of atomic objects and the interaction of the measuring instruments which serve to define the conditions under which the phenomena appear*. In fact, the individuality of the typical quantum effects finds its proper expression in the circumstance that any attempt of subdividing the phenomena will demand a change in the experimental arrangement introducing new possibilities of interaction between objects and measuring instruments which in principle cannot be controlled. (Bohr, 1934, pp.39-40)

Because the measurements are primitive and are a consequence of the measuring apparatus, which itself cannot be fully known except through other measurements, a given change in the measuring apparatus will not necessarily yield a predictable result. The behavior of the resulting measurement interaction "in principle cannot be controlled." Measuring devices cannot therefore be assumed to be in agreement without testing-and their results may be incommensurable, not able to be substituted for each other in a given experimental framework.

One can no longer make assumptions about what a change in the measuring device will bring in the way of different observations. In Bohr's view, measurement becomes an irreducible primitive functionality, one which cannot be avoided altogether by including it in the representational part itself. There is no universal, objective description for what a measurement is, when it takes place, or what is being measured. The measurement problem in physics involves head-on conflict between realists, who believe (in some sense) in objectively correct theories and on the primacy of what the theory represents, and the empiricists, who believe in the primacy of experience and measurements.

Bohr's view carries over into the interpretation of the computational, formal part of a physical model and the relation of mathematics to experience.

When speaking of a conceptual framework, we refer merely to the unambiguous logical representation of relations between experiences....A special role is played by mathematics which has contributed so decisively to the development of logical thinking, and which by its well-defined abstractions offers invaluable help in expressing harmonious relationships. Still, in our discussion, we shall not consider pure mathematics as a separate branch of knowledge, but rather as a refinement of general language, supplementing it with appropriate tools to represent relations for which ordinary verbal expression is imprecise or cumbersome. In this connection, it may be stressed that , just by avoiding reference to the conscious subject which infiltrates daily language, the use of mathematical symbols secures the unambiguity of definition required for objective description. (Bohr, 1954, p.68)

Bohr believed that the essential characteristic of a rigorous scientific theory involved the possibility of agreement within a community of observers. Using symbols, this agreement, this realm of "public" knowledge, could be constructed to achieve high levels of communicability and reliability of interpretation. In this respect the aims of scientific models and mathematical formalisms are the same: to secure to the highest degree the replicability of results across observers. The essence of mathematical formality is the reliable replication of results.

> In his earlier conceptual analyses, Bohr had focused on the proper way of extending classical concepts to, and restricting their usage in, quantum domains. He had accepted the classical meanings of these concepts as essentially unproblematic. His later work forced him to come to grips with the problem of how *any* concept has meaning. Though he never developed a systematic theory, he anticipated some of the key features later developed in Wittgenstein's *Philosophical Investigations*. The meaning of a word is determined by its usage in language, not by the objects it can or may denote. (MacKinnon, 1985, p.115.)

Bohr's analysis of language in physics (e.g. as in his analysis of "physical reality" in his ongoing debate with Einstein (Bohr, 1935a, 1935b; Murdoch, 1987, pp. 168-178)) bears a striking resemblance to Wittgenstein's many observations concerning the nature of mathematical objects (Wittgenstein, 1956) and, in particular, his critique of Cantorian infinities (Shanker, 1987). Like Wittgenstein, Bohr came to believe that properties could not be considered apart from the means of their determination:

> ...we cannot meaningfully ascribe a physical property to an object unless the preconditions for the meaningful use of the predicate are satisfied, and these preconditions are the presence of an appropriate experimental arrangement that is capable of being used to measure the property in question (the weak meaning condition). (Murdoch, 1987, p.176)

Later, seeing difficulties in this position, Bohr strengthened his requirements for the meaningful use of physical predicates:

> ...Bohr added a further precondition for the meaningful ascription of a physical property, which is that an actual measurement of the property in question should have been performed (the strong meaning condition). The mere presence of an appropriate measuring instrument, though necessary, is not sufficient in itself to constitute a well-defined phenomenon in Bohr's technicals sense. ...Bohr made it clear in his Warsaw lecture that a phenomenon is well defined only if a measurement has actually been performed, and the notion of the state of an object is well defined only when applied in a context which includes the performance of the measurement. (Murdoch, 1987, p. 176)

We would do well in our contemporary discourses on physics and computation to follow Bohr's demand that we ground our terms in the physical means by which we would determine their definite values.

The alternative to Bohr's view of the irreducible nature of the observer, the measuring apparatus, and the symbols describing the laws of motion was to attempt to include the measurement process itself in the formal description of the phenomenon. Von Neumann argued that this position would led to an infinite regress as the initial conditions of the measuring device would need to be ascertained by still more measurements.

> ...if we should actually achieve a microscopic rate-equation description of the measuring constraints for the system we are "explaining," we would find that not only the measurement but the system we originally had in mind would disappear, only to be replaced with a new system with an immense number of new initial conditions requiring new measurements. (Pattee, 1979, p.223)

If one also includes in the description the process of constructing the measuring device and bringing the experimental apparatus into the "prepared state," another infinite regress is encountered in the description of the construction and calibration process. If one includes the process of representation and calculation of the results of the measurement, the descriptions of the symbols themselves become part of a circus. Hermann Weyl put it thus:

In existentialism is proclaimed a philosophical position which perhaps is better coordinated with the structure of scientific knowledge than Kantian idealism in which the philosophical positions of Democritus, Descartes, Galileo, and Newton appeared to have found their full philosophical expression.

...When Bertrand Russell and others tried to resolve mathematics into pure logic, there was still a remnant of meaning in the form of simple logical concepts; but in the formalism of Hilbert, this remnant disappeared. On the other hand, we need *signs*, real signs, as written with chalk on the blackboard or with pen on paper. We must understand what it means to place one stroke after the other. It would be putting matters upside down to reduce this naively and grossly misunderstood ordering of signs in space to some purified spatial conception and structure, such as that expressed in Euclidean geometry. Rather, we must support ourselves here on the natural understanding in handling things in our natural world around us. Not pure ideas in pure consciousness, but concrete signs lie at the base, signs which are for us recognizable and reproducible despite small variations in detailed execution, signs which by and large we know how to handle.

As scientists we might be tempted to argue thus: "As we know" the chalk mark on the blackboard consists of molecules, and these are made up of charged and uncharged elementary particles, electrons, neutrons, etc. But when we analyzed what theoretical physics means by such terms, we saw that these physical things dissolve into a symbolism that can be handled according to some rules. The symbols, however, are in the end again concrete signs, written with chalk on the blackboard. You notice the ridiculous circle. (Hermann Weyl, Wissenschaft als symbolische Konstruction des Menchens, Eranos Jahrbuch, Rhein-Verlag, Zurich, 1949, pp.382, 427-428, as quoted in Holton (1988), pp. 210-211)

Bohr's position reasserts the irreducibility of the subject, and the freedom of the subject to choose the means of measuring the world.

... the fact that conscious experience can be remembered and therefore must be supposed to be connected with permanent changes in the constitution of an organism points to a comparison between psychical experiences and physical observations. (Bohr, 1957, p.101)

In a manner analogous to the continual re-evaluation and reformulation of scientific theories, the interactions with the world cause "permanent changes," which continually construct and reconstruct the subject :

...The sign as the locus (constantly interrogated) for the semiosic process constitutes...the instrument through which the subject is continuously made and unmade.... As subjects we are what the shape of the world produced by signs makes us become. Perhaps we are, somewhere, the deep impulse which generates semiosis. And yet we recognize ourselves only as semiosis in progress, signifying systems and communicational processes. The map of semiosis, as defined at a given stage of historical development (with the debris carried over from previous semiosis), tells us who we are and what (or how) we think. (Eco, 1984, p.45)

The natural origins of modelling relations

In our analysis of the modelling relation we assumed at the start the basic capacities of human observers to make measurements, perform computations, evaluate results. But these capacities are themselves evolved capacities. Seen this way, modern physics with its highly self-conscious, highly formalized models of the world is but the latest stage in an evolutionary process which began billions of years ago.

If we look at the evolution of organisms in these semiosic terms, we see that the semantic relations of the organisms' internal states and the world at large have come to be self-constructed and selected through an evolutionary process. How do the modelling relations and epistemic subjects arise in the first place? These questions, which are tied up with the origin of symbol systems at the origin of life, are by no means resolved. To even pose these questions in a meaningful way, however, we need to have a clear idea how to distinguish symbolic processes from nonsymbolic ones, to distinguish what it is we are talking about. We desire a physical theory of symbolic activity.

While the role of symbols in physical modelling has been made explicit for roughly a century, very few scientists have yet taken up questions concerning the physics of the symbols themselves. Howard Pattee and Robert Rosen are among the few theoretical biologists who have taken the role of symbols in biological systems seriously, as their essential organizational characteristic. Their work has involved the explicating the physics of symbols, delineating the role of symbols in biological systems, and drawing out the resulting epistemological implications. This body of analysis is very useful to our purposes here, as a starting point for defining various transformations between symbolic rate-independent structures and nonsymbolic, rate-dependent interactions. The functionalities of computation, measurement, control, and nonsymbolic interactions, which are developed in the next chapter, will be grounded as much as possible in this analysis.

The issues involved are difficult to present because they also implicitly can refer to our status as observers, they are inseparable from *our* epistemic position. The difficulties are similar to those encountered when analyzing logical paradoxes, due to switching between the different interpretive contexts of the paradox. Often in the discussing these matters there is a tendency to jump from an objective, realistic account of the mechanisms involved in the operation of symbols (i.e., the differential equations which describe the physical system) and the conditions under which we as observers use symbols (i.e., in carrying out measurements and performing computations). We need to be consistent regarding which interpretive frame of reference we are using or we will get hopelessly mired in the self-referential complexities. That *caveat* notwithstanding, the objective, realist description of symbolic operation *does* have something to with the subjective, observer-centered one. A human being, organism, or device acting as an observer by implementing a modelling relation can be the object of another observer's physical descriptions. Thus questions asked about symbols in physical and biological systems must also have consistent answers when applied the human observers of those systems. On some level a theory of symbol systems needs to be able to explain itself in order to be internally consistent.

Symbols as special types of physical constraints

Most generally, symbols can be seen as rate-independent constraints on the motions of physical systems, much like the fixed linkages of mechanical devices. Symbols in biological systems are the parts which retain their identity and their interrelatedness amidst the flux of biochemical interactions.

We can use the apparatus of classical mechanics to analyze the role of symbols in physical systems. It should be remembered, however, that the classical interpretation of these dynamical descriptions involves the realist assumption that we have a "largest description" in which all relevant events can be expressed. This, of course, is not true when we leave our formal models to go out into the world to use real world observables to make real world predictions. What follows is an account of symbols assuming a classical omniscience concerning physical laws.

All change in physics can be analyzed in terms of relations which can change and those which cannot. The relations which cannot change place *constraints* on the direction of change. These relations can be further subdivided into those relations which are local, contingent upon particular relations between parts, and those which apply everywhere in the system, the "physical laws" of the system. Properly speaking, the term "constraint" is used to refer only to those unchanging relations which are not the laws. Because these physical laws are everywhere and do not change over time, physical systems without localized constraints are still "constrained" by physical laws, since not all motions are possible, but they have the maximum number of degrees of freedom, or dimensions of motion which are possible. When localized constraints are added, the motion of the physical system is further constrained, and the system loses degrees of freedom.

When we design a mechanical device, we construct it so that stable local mechanical linkages channel the law-governed, "free" motion of the machine in specific directions.

> The tendency of organisms to maintain themselves by their own processes is too obvious to need illustrations. But how are we to decide whether this tendency is to be explained in machine terms, if we are not sure what we mean by a machine? Is just any part of inanimate nature a machine? Sometimes we talk as though we used the concept in this extremely wide sense. Even the physical universe as a whole may occasionally have been called a machine. The trouble is that the same term has also a much more specific meaning, and that, when discussing the topic "Organism and Machine," we are for the most part not aware of this ambiguity. A machine in the more restricted sense is a physical system in which rigid arrangements or constraints compel events to take a certain course. In a well-constructed machine, this influence is one-sided. The constraints of the machine exclude all the possibilities of action which would not be in line with the intended course; but, typically, the constraints cannot be altered by forces which action exerts on such solid conditions [the solid conditions being the rigid members which are the constraints]. (Köhler, 1955, p.328)

The rigid arrangements of parts are called *nonholonomic constraints*. Order can arise both from an imposed nonholonomic constraint and from the free dynamics of the physical laws:

> A simple example will make this clearer. One can easily compel an electric current to take a course which has the shape of a W. For this purpose, it will suffice to conduct the current through a rigid wire, part of which has this shape, and is kept in this shape by being firmly attached to a suitable support. There is nothing in the current as such that favors this particular shape. The form of a W is impressed upon the current only by the described arrangement. It is in this fashion that physical events are forced to follow prescribed ways in man-made machines.
>
> We must next show that the form which a different physical process assumes can also be determined in an entirely different manner. Take this example. A thoroughly flexible insulated wire which forms a closed curve is placed on a smooth and plane surface. At first the curve may be given any arbitrarily chosen shape. This shape will at once be altered if now an electromotive force is induced in the wire so that a current is set up. The shape is changed by the magnetic field of this current, and the direction of the change is such that it enlarges the area surrounded by the conductor. Actually, the conductor may be transformed into a circle the shape in which it circumscribes the greatest possible area.
>
> The difference between this and the preceding example must be obvious. In the present situation, there are no particular constraints by which a special form of the conductor and the current is described. Rather, it is the free dynamics of the system which brings about the change and determines its direction. Obviously, then, we have good reasons for distinguishing between these two factors on which the form of physical events in a system may depend; constraints, on the one hand, and directions inherent in the dynamics of the system, on the other hand. In a given instance, both factors may, of course, be operating. (Köhler, 1955, pp.328-329)

In classical dynamics, the free dynamics are described by universal laws of motion, which means they are time-dependent relations. The rigid local local arrangements of parts are held constant, that is they are time-independent relations. The description of the entire system, of laws of motion operating with the constraints of the parts, is expressed in a set of differential equations describing the rate of change of the state variables of the system, often spatial coordinates, relative to time. The rate-dependent or time-dependent processes can be represented in terms of integrable terms, since each term will involve time. The rate-independent processes, which depend upon nontemporal relationships between the parts of the machine can only be represented in the nonintegrable terms, as inexact differentials.

A given physical system with complex interrelations between its parts can be described by various combinations of these integrable and nonintegrable constraints, but as more of the system is described via the rate-dependent processes, the explicit rate-independent relationships fall out of the description. Similarly, if *all* the relations of the physical system can be described in terms of local rules and interrelations of parts, as in the description of a computer, then there will be no remaining rate-dependent processes. All of the behavior of the system can be described with the nonintegrable, local, rate-independent terms. The nonholonomic constraints are only means we have of altering the motion of the physical system. *If we add enough nonholonomic constraints to the physical system, we can completely eliminate the rate-dependent motions, thereby allowing us to completely specify a trajectory of the system which is independent of rate*. The addition of nonholonomic constraints in the physical system changes the description of the system correspondingly: *If we add enough nonholonomic*

constraints to the equations of motion, we can completely eliminate the terms describing the rate-dependent motions, thereby allowing us to completely specify the trajectory of the system in terms of symbols, as a set of rate-independent rules.

A physical description of the action of symbols can always be described by nonholonomic or nonintegrable constraints. Symbols must be constrained to a few discrete possibilities, yet there must be freedom in the the remaining states that they can assume. Any kind of rule-governed switching can always be described by nonholonomic constraints (Pattee, 1973a). A few examples of holonomic and non-holonomic constraints will serve to make the distinction clearer.

An ideal gas is a completely holonomic system, generally only constrained by its boundary conditions (e.g. a box): every particle has all of its degrees of freedom available to it, and all of the processes which occur are rate-dependent. There is no internal structure to the parts, and they operate independent of each other according to the physical laws which govern each particle. As a result, there will not be any nonholonomic constraints in any of our descriptions since stable, rate-independent structures between the molecules are not present.

A computer, on the other hand, is completely describable as a non-holonomically constrained system. All of the behavior of the system is determined by very special, local, rule-governed state transitions which are independent of time (but not of sequences of state transitions), and completely dependent upon the states of the rest of the system. For this reason, computers can be run at various clock speeds without changing the sequences of states they traverse. This is also the reason that the behavior of a computer can be completely specified by its programmer. In effect the programmer chooses which non-holonomic constraints will be applied such that the desired behavior is achieved. The complete nonholonomic nature of the computer is also the reason that it can be completely described symbolically, in rate-independent terms.

The allosteric action of an enzymes can be seen a nonholonomically constrained process:

> ...all hereditary transmission processes must be executed by non-holonomic constraints. For most biologists, the concept of allosteric device is almost identical with the idea of non-holonomic constraint, i.e. a many structured device which changes its structure upon collision with an outside element (activator or substrate), and thereupon causes a specific reaction to speed up. (Pattee,1968)

The syntactic aspects of transcription and translation, can be described in terms of rate-independent, rule-governed, nonholonomic constraints. The semantic relations of protein folding and morphogenesis cannot be so reduced, being as they are involved with rate-dependent processes. *This is another, physical, way of defining syntactics and semantics: syntactics involve rate-independent, rule-governed relations, while semantics involve the coupling of rate-independent symbols to the rate-dependent world.* In contrast with the computer, the allosteric enzymatic device is not completely specifiable in terms of its nonholonomically constrained action, since the folding of the enzyme and its stability in solution are both rate-dependent processes.

The rudiments of symbolicity: analysis of a simple switch

The simplest imaginable discrete, symbolic function is the operation of a two-state switch (Pattee, 1973a). Symbols switch a physical system from set of one dynamical states to another. One can easily identify discrete states and construct a state transition table for a resettable switching device (the nonholonomic description), but finding a continuous model or a dynamical system which implements the switching function is more difficult. Note that the very terms with which the two types of constraints are described are different. An example of a continuous model of a switch is Zeeman's catastrophe machine (Zeeman, 1972). The device implements a cusp catastrophe using two rubber bands and a disk with a fixed pivot. The system has two stable states as well as history-dependent characteristics. This is

a system driven by external forces and involving irreversible, dissipative processes (the rubber bands). Were it implemented as a reversible, dynamical system, it would lose its switching function:

> If one constructs this machine with great care, using fine bearings and elastic springs, so that there is very little friction, it will not return to equilibrium very quickly after being triggered but will undergo damped oscillations. Under these conditions, we must restrict the interpretation of S_0 to a region very close to the equilibrium point, otherwise the trigger threshold will be phase or time dependent. In other words, the switch will be sensitive to a great variety of input triggers that are not independent of the detailed dynamics. Such a device that does not effectively suppress the continuous dynamics would not be interpreted as a reliable switch. (Pattee, 1973a)

In order for the switch to have stable states, there must be at least two attractor basins in the state space of the dynamical system. In order for the result to be independent of time, the system must settle into one of two distinguishable states.

> This illustrates the fundamental physical condition imposed on all switches, or more generally, on all logic, that for every discrete switching event (or equivalently, for any record, measurement, classification, or decision process) there must be a corresponding dissipation of energy. (Pattee, 1973a, p.17)

Thus we have a linkage between the discrete nature of the symbolic states of the switch, the necessarily dissipative nature of the attractor basins which form the states, and the irreversibility of behavior which results.

Measurements and controls involve nonholonomic constraints

The process of measurement can similarly be analyzed in terms of types of constraints, as a coupling between rate-dependent processes and rate-independent symbolic structures.

> The classical scientific concept of measurement requires a distinct physical measuring device that selectively interacts with the system being measured, resulting in symbolic output, usually numbers. (Pattee, 1985, p.269)

The interaction with the system being measured may be indefinitely complex, but the result must be one of a finite number of distinguishable symbols. Each of these symbols is an equivalence class of many possible interactions. Thus each measurement implements a many-to-one mapping from an indefinitely large number of interactions to one equivalence class, which is represented by the output symbol.

> *...this classification property of measurement is an epistemological necessity*. Without classification, knowledge of events would not be distinguished from the events themselves, since they would be isomorphic images of each other. (Pattee, 1985, p.270)

Moreover, "all biological information originates from measurements," from the actions of enzymes recognizing substrates to the mechanisms of natural selection (Pattee, 1979, p.223). Pattee has outlined the general structural and functional features of a measuring device:

> (1) Measuring devices are localized, isolatable, resettable structures with repeatable actions.
> (2) Measuring devices have no intrinsic output actions, but may be triggered to simple actions by specific input patterns (nonholonomic constraints).
> (3) Measurement constraints obey all physical laws, but are not derivable from laws (generated by system function).
> (4) Measuring devices are constructed sequentially under the control of linguistic constraints, but a complete set of measuring devices is necessary to read linguistic strings (semantic closure).
> (5) Measuring devices execute a many-to-one mapping from complex input patterns to simple output actions (classification).
> (6) Measuring devices do not occur in isolation, but form functional, coherent sets within a system.
> (7) The value and quality of any measurement is a system property determined by the survival of the system in which it functions.

(8) Beyond these properties, the domain of input patterns, the range of output actions, the choice of mapping, and many other aspects of measuring devices are largely arbitrary." (Pattee, 1985, p. 277)

The description of a measuring device necessarily involves the use of a non-holonomic constraint:

> This ability to recognize complex input patterns and, as a consequence, execute a simple action requires physical constraints of a special type. Since the many-to-one mapping is arbitrary, the constraints must abitrarily couple the *configurations* available for fitting the input pattern to the *motions* of the device that produces the output actions. In physics these are called nonholonomic, or nonintegrable constraints. A holonomic constraint is a restriction on the configurations of a set of particles, such as occurs in forming a crystal from a solution of molecules. This freezing-out of the configurational degrees of freedom necessarily freezes out the corresponding motions of crystallized molecules, so that we see the constrained system as a rigid solid. A nonholonomic constraint may be defined as a restriction on the motions of the particles without a corresponding restriction in the particle configurations. In other words, a formal expression of a nonholonomic constraint appears as a peculiar equation of motion for selected velocity components, where certain configurational variables serve as initial conditions. However, we cannot eliminate any configurational variables of the system by using these relations because of the nonintegrability of the equations of constraint. This results in a flexible or allosteric configuration. (Pattee, 1985, p.271)

Controls, which couple the rate-independent symbols to rate-dependent actions also must involve nonholonomic-constraints because of the nonintegrabilty of the rate-independent, symbolic dependency. Computations, as we have noted above, are physical processes which can be *completely* reduced to nonholonomic constraints, such that there are no rate-dependent quantities left in the description. Measurements and controls can be distinguished from computations in that they cannot be fully so reduced; in either case to do so would entail removal of the rate-dependent part of the relation. Figure 4.3 encapsulates these relationships:

computation rate-independent to rate-independent coupling	**control, construction** rate-independent to rate-dependent coupling
measurement rate-dependent to rate-independent coupling	**physical laws, equations of motion** rate-dependent to rate-dependent coupling

Figure 4.3. Types of couplings between rate-dependent and rate-independent processes. All constraints that describe a rate independent process (i.e. computation, measurement, control) are nonholonomic constraints. IN this classical view, one can always replace rate-independent terms with more complex rate-dependent ones. Thus, computation can be seen as a series of either measurements or controls, depending upon whether the dependent or independent term is replaced. The reverse is not generally true in this scheme, i.e. that measurements and controls can be seen as computations.

This explains why all computations can be seen as a series of measurements or a series of controls, but this is not always the case in reverse. Given a measurement or control, it is not necessarily always possible to reduce the rate-dependent part of the relation to a rate-independent structure. However, it always is possible to go the other way, to express a rate-independent structure in terms of rate-dependent equations of motion. Thus computations can always be described in terms of measurements and/or controls, measurements and controls can always be described in terms of the rate-dependent laws.

A concrete example will be helpful. Consider the inclined tube-track with assorted steel balls running down it illustrated in figure 4.4.

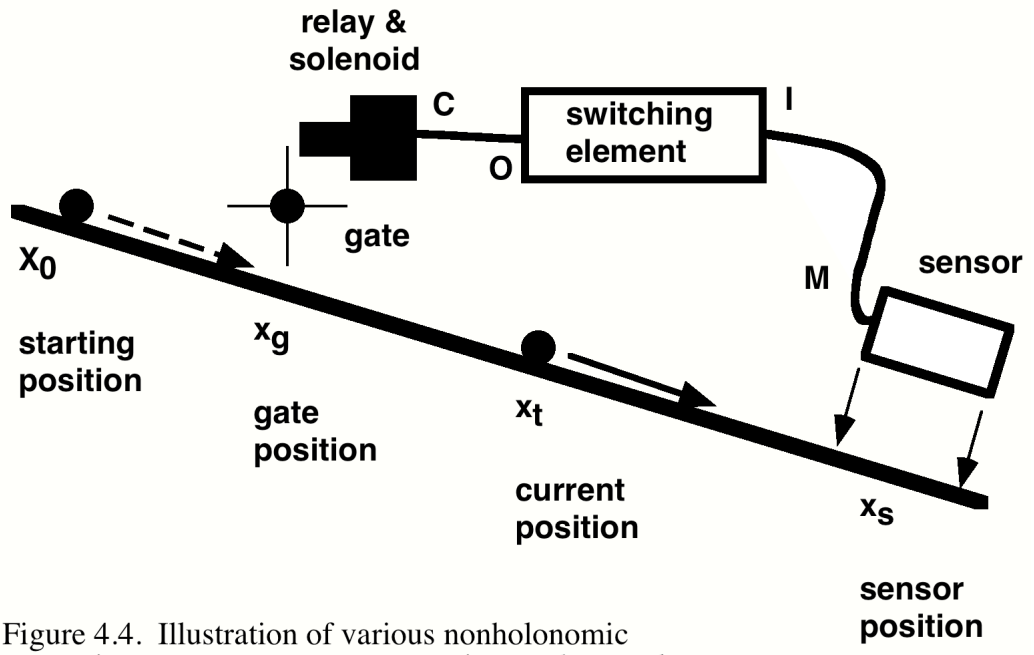

Figure 4.4. Illustration of various nonholonomic constraints: measurement, computation, and control.

The free dynamics of the system are expressed by the equations of motion, which express the gravitational attraction of the balls and the earth. The boundary condition is the track, which we will assume is perfectly straight. The initial conditions are the initial positions of the balls on the track and their respective masses. We will now add three kinds of nonholonomic constraints to the picture, a measurement process, a control process, and a computational process.

Towards the top of the track we will add a gate which is open or closed via a solenoid and a relay switch. If the relay switch terminal is set high, at around 1 volt, the solenoid opens the gate and allows balls to pass, if it is set, low, at around 0 volts, the solenoid will close the gate. The threshold of the relay switch is somewhere between 0 and 1 volts, so exact voltages do not matter, but whether it is high or low does matter. A ball will roll down the track until it encounters the gate. If the gate is open, the ball will continue rolling unimpeded, according to gravity. If the gate is closed the ball will stop completely. When the gate is opened the ball will resume its motion under gravity, but this time starting from a state of rest. Here the rate-independent binary voltage is coupled to the rate-dependent motion of the ball at the point of the gate. We will use the symbols **a** and **b** to represent the high and low voltage states in our equation of constraint and **C** to be the state of the controller relay. Let X_0 be the initial position of the ball on the track, X_t be the current position of the ball on the track and X_g be the position of the gate on the track. Thus the equation would be:

$$dx/dt = 0, \quad \text{if } \mathbf{C}=\mathbf{a} \text{ and } X_t = X_g$$
$$dx/dt = f(X_0), \quad \text{if } \mathbf{C}=\mathbf{b} \text{ and } X_t = X_g$$

Note that the gate is a dissipative constraint; the ball loses energy to the gate and stops completely. For the sake of simplicity, we will assume this is instantaneous.

Towards the bottom of the track we will add a sensor which measures the speed of the balls rolling through that section of track. We will also assume that this measurement occurs instantaneously. Note that this is also a dissipative constraint, that some energy must be imparted to the sensor to make the measurement, but we will neglect this. The sensor will output a low voltage of approximately 1 volt if the ball's speed exceeds a certain threshold, V_c and approximately 0 volts if it is less than the threshold.

The output value of the sensor stays in its current state until the next ball comes by when it will perform a new measurement. Thus the sensor's output reflects the speed of the last ball. Here the rate-dependent motion of the ball is coupled to the rate-independent output voltage of the sensor. If we represent the output state variable of the measuring device as **M**, the low voltage output as **c** and the high voltage output as **d**, and X_S is the position of the sensor, then we could write the constraint as:

M = c if $dx/dt > Vc$ and $X_t = X_S$
M = d if $dx/dt < Vc$ and $X_t = X_S$

Between the gate and the sensor we place a switching gate which takes the output voltage of the sensor (low or high volt) as an input and outputs a low or high voltage to the relay controlling the gate. This switching element, which implements a Boolean function, can be completely and adequately described by the rate-independent mapping of the binary voltage levels. Here we will write the constraint for a mapping of low input to high output and high input to low output, where **O** is the output of the switching element and **I** is its input.

O = a if **I = c**
O = b if **I = d**

Note that all three of these devices could be described by the rate-dependent equations of motion for the mechanisms (which could be quite complex, involving dynamical and electromagnetic state variables), but that not all of them could be completely reduced to rate-independent relations. The motion of the ball through the sensor's segment of track or its movement through the gate could not be so reduced because it involves continuous motion. This incommensurability of rate-independent and rate-dependent descriptions underlies the irreducibility of the processes of computation, measurement, control and nonsymbolic interaction defined in the next chapter.

It is also important to note that descriptions using the nonholonomic constraints involve an enlarged set of state variables, since we now need the symbolic states **a** ,**b** , **c**, and **d** for the new state variables of **C**, **O**, **I** and **M** in our description. The observables that these symbols represent are also effectively new observables corresponding to the symbolic states of the system having the states "low input," "high input," "low output," and "high output."

Why constraints cannot be logically deduced from physical laws.

Can the foregoing analysis of symbols aid in the development of a description of their origins? In other words, can we logically derive the nonholonomic rate-independent rules from the rate-dependent equations of motion? In the above example this would correspond to deriving the nonholonomic equations of constraint from just the equations of motion. This bears on the question of whether we could, by computing the equations of motion of the universe, derive emergent higher-order invariant properties. If we could somehow do this, we would in effect be creating new observables through through logical operations. As Pattee and Rosen observed, this appears to not be possible, at least using the discrete symbolic descriptions of our contemporary mathematics, due to the nature of those descriptions.

> Can our physical description of control constraints help clarify this problem or suggest a way out? We have emphasized the idea of a constraint as an alternative description of a physical system which can also be described in principle by microscopic laws of motion. This is therefore true of control systems in general. What is there, then, about the physical description of the origin of constraints that causes so much difficulty? In most physical theories we define the system we are talking about by fixing a number of degrees of freedom and then expressing the forces or interactions on these variables. It is easy to conceive of these forces being gradually removed so that we can see how

> the behavior of the fully interacting system grows continuously out of the non-interacting particles. We treat equations of of constraint in quite another way--they are either present or absent--and there is no gradualness about it. This is because the equations are an alternative description. In other words, when we choose to use an alternate description, there can be no gradualness about it. We cannot pass from one description of a system to another in a continuous way. I believe this is the fundamental difficulty in describing the origin of new physical levels of control. All new hierarchical levels of organization require an alternative description [Pattee, 1971; Rosen, 1969], and consequently there is no obvious way to describe gradual transitions between descriptions. Yet it is hard to believe that this is all nature's fault. The problem appears to reside largely in the nature of language itself which requires a discrete, fixed set of grammar rules which are not subject to continuous transformation. Again we may look at the physical interpretation to see if there is some way out. (Pattee, 1973, p.44)

> Now how can we go about actually constructing a good dynamical theory of evolutionary novelty? As Pattee has so abundantly recognized, the kind of dynamical modelling necessary for a good theory of evolution (and of evolutionary novelties) must carry within itself the possibility of effectively *generating*, from the intrinsic dynamics of the system, *meaningful new state variables* appropriate for the description of new functions emerging as a result of the interaction between the evolving system and its environment. (Rosen, 1973, p. 135)

For the generation of emergent non-holonomic constraints we must either abandon our present deterministic, rule-based, symbolic mathematics or look outside of formal solutions for our answer. Descriptions which generated "new meaningful state variables" in the form of entirely new symbols appearing in the description would no longer be formal constructions, because their appearance would involve new, undefined symbol primitives.

The result of these considerations leads to the conclusions that *fundamental emergence cannot be fully described by formal systems and that purely formal devices do not in and of themselves generate emergent behavior*. No new state variables appear that were not in the original model. No new primitive symbols arise. What can arise are only combinations of the symbol primitives in the model from the start. We will discuss this issue throughout the rest of this dissertation. Rosen's concept of emergence as the inevitable deviation of the behavior of the natural system from a given formal model of it (Rosen, 1985a,b, 1987) flows from this inability of the formal model itself to generate fundamental novelty.

The only way to escape these conclusions is to deny the distinction between rate-independent and rate-dependent descriptions by asserting the absolute reality of either the rate-independent symbols or the rate-independent laws. The world is either seen as fundamentally discrete and symbolic or it is seen as fundamentally continuous. If one holds either of these views then the symbol-matter distinction disappears. Both of these positions, the ontologies of the computationalists and the ecological realists in the cognitive science debate respectively, have been extensively critiqued by Pattee (1987, 1988, 1989).

Complementarity of laws and constraints

The foregoing analysis leads us to a complementarity of descriptions, those utilizing the nonholonomic constraints (local rules) and those utilizing only the free dynamics (universal laws). These descriptions are not derivable from each other, because the nonholonomic constraints are not expressed in the same terms as the laws.

We have shown in the last section that the rules cannot be automatically deduced from the operations of the laws. But how do we get from laws to rules in *our* descriptions? How do non-holonomic constraints ever appear in *our* equations? The answer is that an external observer must recognize a stable pattern temporally, via a rate-dependent process analogous to measurement and encode that pattern into a rate-independent set of symbols constituting the nonholonomic constraint in the equations of the system. This is basically how new mathematical concepts must arise, from nonmathematical means.

Let us suppose we run a simulation of a dynamical system, one in which only physical laws are represented. As the simulation runs, we begin to recognize higher order patterns and invariant relationships or structures. At this point we change the simulation to stabilize the invariant relationships by turning them into non-holonomic constraints. We write the rule-governed relations into our

equations. What has happened? As external observers we have made an inductive, empirically-based leap in recognizing the invariant pattern and we have changed the nature of the simulation from without. The observer has brought to the simulation an observable which was not distinguished in the simulation itself, namely the higher level pattern. This observable is a static pattern rather than a dynamic state. By stabilizing our observed, time-dependent, *temporary* pattern into a time-independent structure, we have thereby guaranteed that the configuration will not become unstable. We have shifted our simulation from temporal processes to nontemporal structures. At the point that we do this, the two simulations have the same behavior. But we now have two simulations which will diverge in behavior at some point in the future when the pattern becomes unstable in one but persists in the other. The organon program of Minch (1988) involves precisely this process of forming stable structures out of dynamical interactions.

Seen in this way, rate-dependent laws and rate-independent rules are irreducible, complementary descriptions involving different descriptive terms. This complementarity supports a non-reductionist account of function: while all physical interactions must be consistent with laws, the functional rules implemented through the action of nonholonomic constraints are not derivable from the laws.

This distinction between necessary and contingent realms is what distinguishes classical physics from biology: classical physics depends upon necessary, law-like relations, while biological evolution can only be explained in terms of initially arbitrary structures which become more ordered through selection. Classical physics takes the position of an observer for whom all conditions are knowable, quite like that of the correspondence theory of truth and the perspective of symbolic reference. Evolutionary biology adopts the perspective of the organisms, operating under limited information and having access to the world only through evolved sensory channels. Seen in this way, classical physics becomes the science of necessary order in the world, evolutionary biology the science of contingent order. Consequently each favors a different view regarding the origins of order: classical physics favoring an order-from-order ontology, evolutionary biology preferring an order-from-noise epistemology.

Emergent semantics and the origins of constraints

The irreducibilities outlined above mean that we can break down operations involving symbols into discrete types of operations depending upon how the symbols are coupled to rate-dependent processes. For devices with emergent semantics we need to form new couplings between symbols and rate-dependent processes. How can this be done? It cannot happen within a completely symbolic, rate-independent realm, as we have argued above, because their is no connection to rate-dependent processes. We need a process of construction in which heritable, rate-independent symbolic constraints constrain the rate-dependent formation of the coupling structures. In the immune system DNA constrains the construction of antibodies which then function as measuring devices, coupling a rate-dependent steric interaction to a rate-independent activated form, which serves as a signal to other immune cells that an antigen has been recognized.

Once an organism or device possesses the means of reliably constructing its own couplings between rate-independent and rate-dependent processes then any combination of these processes can be implemented. Not only can rate-independent syntactic constraints be constructed, but also semantic constraints linking rate-dependent and rate-independent processes. *In a sense the organism or device becomes both syntactically and semantically programmable*. The organism or device is now in a position to evolve through selection the symbol-world relations necessary for survival. The organism or device has achieved a degree of epistemic autonomy, or in Pattee's terms, *semantic closure* (Pattee, 1982; Pattee, 1985). Similar to this concept of semantic closure is the conception of the operational closure of the nervous system advocated by Maturana and Varela (1987). While the crucial distinction between rate-dependent and rate-independent processes is not made in the case of operational closure, the essential circularity of the structure-perception-action-performance-construction relations can be found here.

The modelling relation discussed above is also connected to this condition of semantic closure. Processes of measurement, computation and the construction of the measuring devices themselves can be seen in terms of these rate-dependent/rate-dependent linkages. All are necessary for the modelling relation to be implemented, for coupling our symbols to the world in various ways. When they are all present, there is a situation where we can have a self-constructing epistemic subject in which those perceptions and measurements which are useful can be selected out of many possibilities.

We will use these irreducible functionalities and the concept of semantic closure to analyze different types of constructible devices and to outline a strategy for constructing devices which construct their own epistemic relations to the world.

Chapter 5 Primitive symbol-matter transformations

Having considered the roles which symbols play in biological evolution, in the modelling relation, and in classical mechanics descriptions, we saw that there are four basic types of relations symbols can implement: *computation*, *measurement*, *control*, and *nonsymbolic interaction*. In this chapter we will define these relations more precisely, since they will form the functional primitives of the various device types we will analyze later on. Once this is accomplished, we can clearly analyze the structural preconditions for devices which adaptively construct their own perceptual, cognitive, and behavioral structures. We can then compare various types of devices, noting their relative capacities and limitations.

A functional framework for the analysis of devices

The significance and relevance of these functionalities will be immediately understood if we first introduce the context of their use. Consider a general framework for the construction of devices made up of sensors, coordinative computations, and effectors in place of their organismic counterparts: perceptions, perception-action coordinations, and actions. In place of perceptions, coordinations, and actions, we will have the primitive functionalities of measurements, computations, and controls. The perception-cognition-action system sketched out in chapter 3 and figure 3.1 thus can be recast in terms of measurements, computations, controls. Nonsymbolic interactions will be seen as part of the environment. The framework can then be explicitly partitioned into dichotomous symbolic and nonsymbolic processes, of coded and uncoded mediations (figure 5.1). In this interpretive framework measurements and controls define the relation between symbols and the world at large. Dotted arrows indicate nonsymbolic interactions, while solid arrows indicate symbolic or type-based interactions.

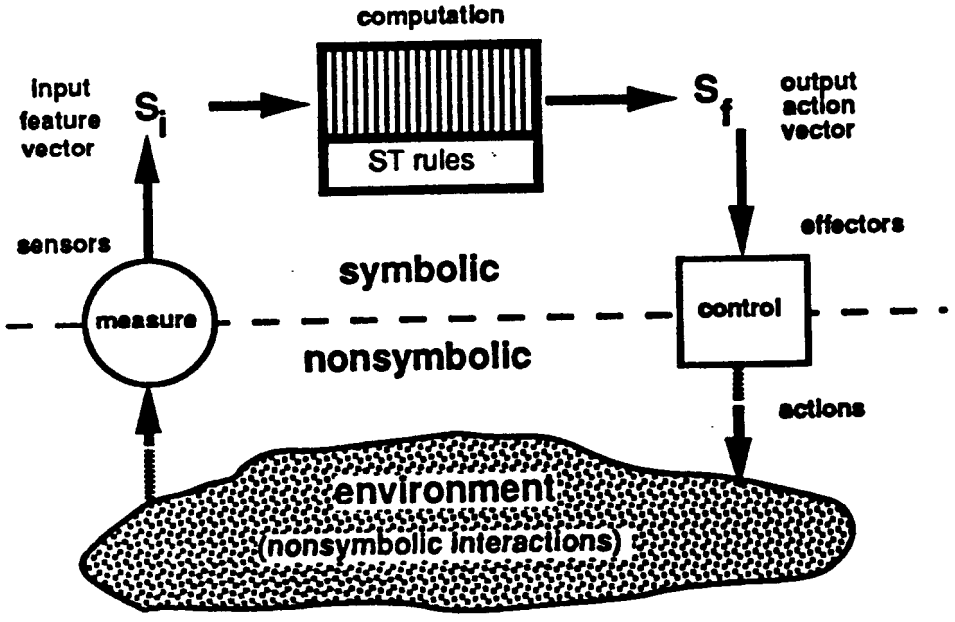

Figure 5.1 Symbolic and nonsymbolic realms

Seen in this way measurements and controls implement semantic relationships between symbolic and nonsymbolic events, while computations implement syntactic relationships between symbols. Those processes selecting which computations are to be performed and which semantic relations are to be utilized are pragmatic relations.The three axes are illustrated in figure 5.2:

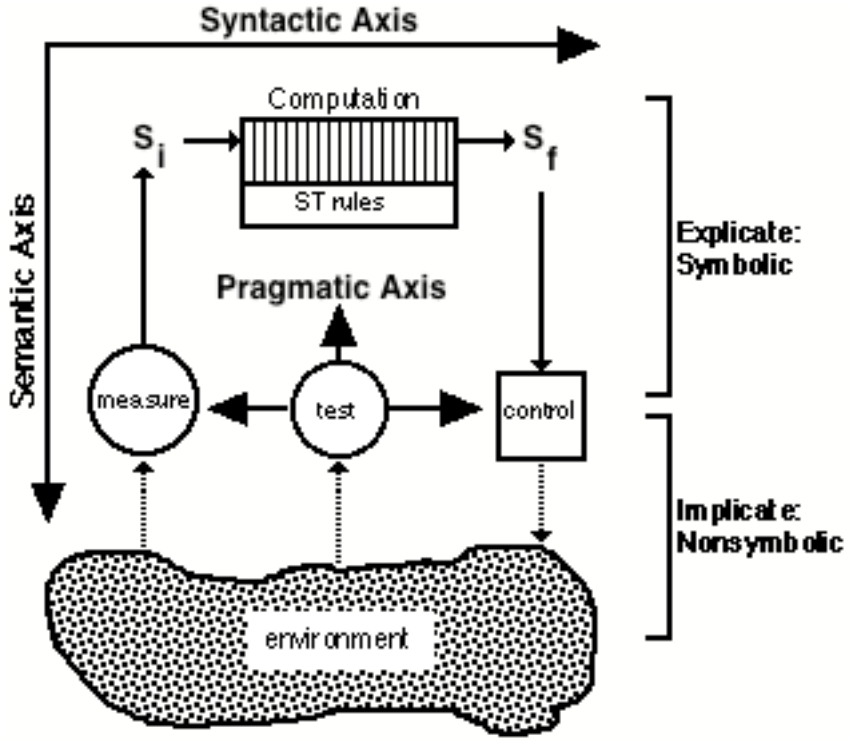

Figure 5.2 Semiotic relations in adaptive devices.

In this and other schematics, for the sake of clarity, each functionality potentially represents more than one process. Thus "measure" could encompass multiple sensors and multiple observables, "computation" could encompass any input-output function of distinguishable observable states to distinguishable controls, and "control" could similarly refer to multiple effectors. A given organism or device could also have multiple semantic-syntactic loops of this sort, each acting independently of the others. For simplicity only the properties of single loop organisms and devices will be discussed.

Symbolic and nonsymbolic realms

A *symbolic process* is generally understood as one which can be described solely by the interaction of tokens, categories, or types (e.g. Haugeland, 1980, 1985, ch. 2; Pylyshyn, 1984, ch. 3; Newell, 1980; Newell & Simon, 1981). A *nonsymbolic process* is any process which is not symbolic, which cannot be completely be reduced to interactions between symbolic types.

Conceptually, the symbolic/nonsymbolic distinction underlies those of software and hardware, mind and body, formal and physical, noumena and phenomena. In the philosophical and artificial intelligence literature, the distinction has been variously framed in terms of levels of "semantic" and "symbolic" processing vs. functional architecture (Pylyshyn, 1984), digital vs. analog (Haugeland, 1980, 1985), cognition vs. subcognition (Hofstadter, 1982), program-receptive vs. program-resistant features (Gunderson, 1985), controlled vs. autonomous action (Dennett, 1984, ch. 3). In physics the distinction is between measured and unmeasured, between measuring and calculating, between implicate and explicate interactions (Bohm, 1980). In design it is between designed and haphazard properties, between specified and autonomous behavior.

We will adopt the operationalist, interactionist attitude of systems science and cybernetics, except where noted, for deciding whether a process is symbolic or nonsymbolic. *Whether we recognize a process as symbolic or not is dependent upon our relation as observers to that process.* "Symbolicity" is

not an inherent property of either the real world object alone or the observer alone; it is a property of their interaction. That we are not utilizing a classical, realist,"objectivist" conception of symbols cannot be overstressed.

Transformations between the symbolic and the nonsymbolic

Once the symbolic-nonsymbolic distinction is made, transformations between the various kinds of processes can also be distinguished and their functionalities analyzed. Four basic transformations can be distinguished: computation (symbols to symbols), measurement (nonsymbolic interactions to symbols), control (symbols to nonsymbolic interactions), and nonsymbolic interaction (nonsymbolic interactions to nonsymbolic interactions). Figure 5.3 summarizes these transformations.

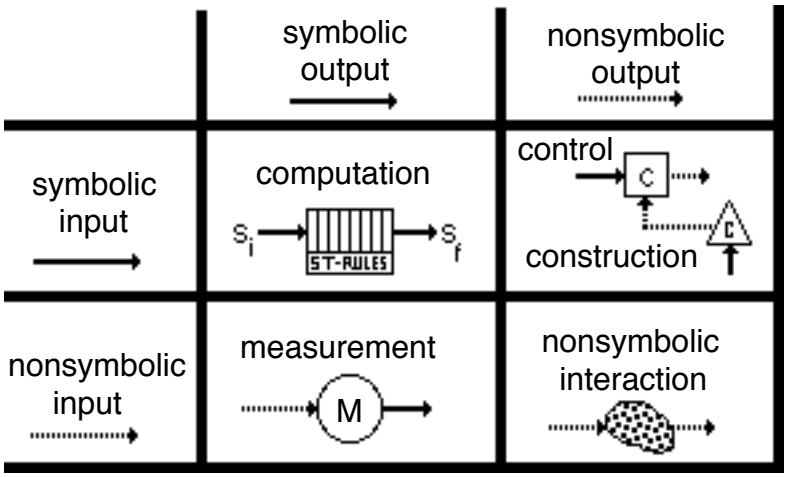

Figure 5.3. Basic transformations between the symbolic and the nonsymbolic.

In the terms of the last chapter, computations can be completely described by the action of completely rate-independent constraints, while measurement and control can be described by mixtures of the two kinds of constraints.

> ...a measuring device is a non-integrable constraint that acquires information from a rate-dependent dynamical system while a control system is a non-integrable constraint that influences the rates of a dynamical system. The common essential requirement of measurement and control constraints is that they must couple two systems, one described as rate dependent and the other as rate independent. (Pattee, 1979, p.223)

Nonsymbolic interaction would be behavior whose description could involve only rate-dependent processes (no non-holonomic constraints could be found to describe the behavior).

Two other process subtypes will also be considered: construction and selection. Construction (symbols to nonsymbolic interactions) is a special form of control: when control is applied to the organism or device itself. Selection (nonsymbolic interactions to symbols) is a special form of measurement: a measurement upon which the structure of the organism or device itself is dependent. In biological evolution selection is the effect of differential survival on the genes present in a given population; thus the genes remaining in the population are the symbolic output of nonsymbolic interactions.

> Natural selection ... is normally defined in terms of the coupling of between genotype and phenotype where the concept of genotype requires informational selection driven by the rate-determined dynamic selection at the phenotypic population level. (Pattee, 1979, p.224)

In contemporary learning machines selection takes the form of measuring the machine's real world performance and altering the device's internal computational structure accordingly.

The observational frame

In the last chapter we discussed the description of various types of dynamical constraints, involving various combinations of rate-dependent and rate-independent constraint relations. Completely symbolic, syntactic constraints are those which can be described completely in terms of rate-independent relations. Semantic processes are those which couple rate-independent symbols and rate-dependent processes. Completely nonsymbolic processes are those for which no non-holonomic constraints can be found. In the analysis of the last chapter we assumed a classical perspective, that all the physical laws, boundary conditions, and constraints for the system were completely known. It should be recognized that this criterion of whether we are dealing with a symbolic or a nonsymbolic operation involves some measure of subjectivity, since recognizing the existence of a nonholonomic constraint and incorporating it into the description is up to the observer. In particular, whether one will recognize a rate-independent, rule-governed, *symbolic* linkage through one's observations of a physical system depends on the observables chosen, on our *observational frame*. Once this frame is fixed, however, the state transitions we observe *can* be unambiguously partitioned into rule-governed and non-rule-governed operations, depending on their observed behavior.

In this chapter we will analyze the functionalities of measurement, computation, control, and nonsymbolic interaction *from the point of view of an external observer, rather than from a classical point of view*. Our classification of the functionalities will be based here on the observed behavior of the physical system, not on a fundamental set of theoretical relations.

Definition of an observed state

The notion of an observed state is fundamental to all attempts at clearly describing the situation of the observer examining a physical system. "By a state of a system is meant any well-defined condition or property that will be recognized if it occurs again" (Ashby, 1956, p.25). A state is a relationship between the observer and the physical system being observed. It is not an inherent property of either the observer or the observed physical system, but a property of both. The "well-defined condition or property" is defined by the measuring device used to observe it. The observer's states are thus implemented by the process of measurement; *they do not exist prior to the measuring process which brings them into being*. They represent a classification of the measuring device's interactions with the physical system. This classification is the categorization into one of two or more discrete symbolic types.

The operationalist, observer-centered, phenomenalistic concept of state used here is distinct from the classical, "objective" conception, in which physical states exist independent of their observation. Here, as in the Copenhagen interpretation of quantum mechanics we consider a state to be a joint property of the observer-observed system combination, and it is a free choice by the observer which sets of states or observables will form a particular observational frame.

Symbols and the property of type

A type property is implemented when the observer observes the state of the physical system. An example of a type property is the recognition of the numeral "**1**" in a sequence of numerals. Once the individual token numeral is correctly categorized as a "one" as opposed to a "two" or a "three" or any other possible category of numeral, all of its physical properties become irrelevant to its function. Its classification by a particular measuring device becomes its only relevant property, enabling one to describe all that is relevant about the symbol by referring only to its type.

The type property is a "symbolic" property; symbols are identified by an external observer as symbols by virtue of being vehicles for the transmission of this type property. There are indefinitely many nonsymbolic properties associated with every physical system; as many as one cares to measure (Ashby, 1952, p.15); but symbolic properties are restricted to the finite set of recognizable types. It is this distinction which allows the possibility of semantic open-endedness and emergence in physical systems with nonsymbolic components (i.e. as in measurements or controls), while those having only the syntactic properties of type remain closed and finite (Rosen, 1985). This idea will be developed further in the final chapters of this dissertation.

How does an observer of a physical system, organism, or device distinguish between those observed state transitions which are purely symbolic and those which are not? When is the state transition only dependent upon the type property of the input symbol? In observing the physical system, the observer has defined states of the observer-system relation and if the physical system is employing a type-determined transformation, then some measuring device will yield states that are close enough to those of the natural system to exhibit the same behavior over some finite period of time. If type alone is involved then it will be possible to find observed state transitions which are rule-governed, deterministic, one successor state for every predecessor state. Symbolic interactions are those based upon the recognized "type" of the symbol vehicle rather than "continuous", non-discrete, undefined, or undescribed interactions. Rule-governed, symbolic interactions imply deterministic transitions, since only one property of the input symbol vehicle, type, is involved.

As a result of the foregoing analysis, it is possible for an observer to distinguish between the observed states implementing purely symbolic interactions from those implementing nonsymbolic ones. The system is observed through the observer's frame over a period of time sufficient to see the same states multiple times. Either each state will be immediately followed by one unique successor state or it will have multiple successor states. The former will appear as completely deterministic observed state transitions, the latter as nondeterministic observed state transitions. In the case of deterministic state transitions, the observer can formulate a rule relating the predecessor state and the successor state. This is related to the recognition by an observer of a rule-governed, completely rate-independent constraint discussed in the last chapter. The observed behavior of the system in that state can be replaced by the rule. In the nondeterministic case a rule cannot be found which will completely replicate the state-transition behavior. If the observer believes that the physical system under observation is indeed implementing symbolic interactions, but the observed state transitions are not deterministic, then the observer can redefine the states of the system by using other measuring devices. Existing states can also be merged by dropping observable distinctions in the observational frame. However, unless the observer can specify a set of observables which will yield a deterministic state transition, it will be assumed that, given the observer's observational frame, the relation between the observed states is a nonsymbolic one.

We are now in a position to discuss the primitive transformations between the symbolic and the nonsymbolic.

Computation

Computation is a purely symbolic operation involving only symbolic type properties and syntactic operations. Computation is the transformation of an initial (symbol) state into a final (symbol) state in which only the description of the states themselves is sufficient to deterministically replicate the final state. Computation can be seen as the mapping of input symbol strings representing the initial state onto output symbol strings representing the final state via rewrite rules (implementing a state transition rule).

Computation can also be described wholly by nonholonomic constraints (chap. 4); physical laws need not enter into the description of the behavior of a physical system implementing a computation.

Determinate nature. The functionality of computation is the transition from an initial state to a final state by virtue of only the type property of the initial state and not of extraneous interactions with the

world, which would involve non-type properties. This implies that the transitions of symbolic states to other symbolic states is unique, that one symbol state will give rise to one and only one final symbol state, because the initial state has one and only one type designation and the transition depends only upon type designation. When the physical system is constructed in such a way that the transitions can be described solely by means of the type-type transitions over a given period of observation, then it can be said that the interactions are purely symbolic in character. A computation can be said to take place if the states of the system are defined so as to permit observation by a community of observers, the state transition rule is defined, and the computation reaches its final state within an observationally relevant time period.

Identifying computations in natural systems. Computation is therefore not a property of the natural system by itself, but of our means of interacting with it and observing it.

To identify a computation in a natural system, it is necessary to distinguish discrete states of the system by specifying an observational frame, identify deterministic state transition rules, and verify that the behavior of the observed system corresponds to those rules. *These are the same conditions that would be needed if one were to utilize the prospective natural system to carry out the presumptive computation. It is gratuitous to declare that a computation is occurring in a physical system if one does not have the means of actually utilizing it as a computation.*

For the computation to be programmable, the observer must be able to bring the physical system into the appropriate initial state. Otherwise, the computation is nonprogrammable. In this case the observer has no control over which computation is being performed; nevertheless the operation is still a computation as long as the observer can observe the initial state and has access to the (deterministic) state transition rules.

If the observer cannot observe the input state (i.e. if it is undefined or unobserved) or if the observer cannot construct a deterministic mapping between an observed input state and the observed output state, then, relative to the observer, the operation is a measurement (see below). If the observer can observe the initial state, but not the final state, or if the observer cannot construct a deterministic mapping between the initial state and the final state, then there is no longer the type-type transition and an interaction involving some properties other than type has occurred.

Error. Contemporary computers realize computations when they are behaving according to their specifications. Occasionally even contemporary digital electronic computers make errors or fail completely. When these devices have deterministic, replicable behavior, they are carrying out computations. When their behavior deviates from their expected behavior, they are no longer implementing the type-type transformations. Given enough time all physical systems will eventually deviate from any finite, deterministic model (Rosen, 1985), but devices can be constructed to be reliable over limited, humanly-relevant periods of time. In order to accomplish the functionality of computation, a process must be deterministic and replicable over the relevant period of observation. The process must be reliable enough so that the possibility of error is rendered irrelevant to the particular computational task at hand.

State-transition rules. The mapping of initial state (input state) to final state (output state) can be described by a state transition rule, which prescribes for all possible, allowable defined inputs what the output state will be. The state transition rule can be a look-up table relating input to output state, or it can be an algorithm, possibly with many intermediate substeps, which implements the necessary transition.

Compositions of computations are computations. Compositions or couplings of computations produce other computations.

> A fundamental property of machines is that they can be coupled. Two or more whole machines can be coupled to form one machine; and nay one machine can be regarded as formed by the coupling of its parts, which themselves can be thought of as small, sub-, machines. (Ashby, 1954, p.48)

As long as each step is deterministic, i.e. as long as each input-output relation is a function, then the total input-output relation will be a function.

Complete formal description. Because of their deterministic behavior it is possible to completely describe the structure of computational processes in formal terms.

> *Finite state machines* or *finite automata*...are machines which proceed in clearly separate 'discrete' steps from one to another of a finite set of configurations or states....Because of their peculiarly limited, finite nature, the structure of these machines is easily described *completely*, without any ambiguity or approximation. (Minsky, 1967, p.11)

This criterion of complete, unambiguous description is the essence of formality, but this can only be achieved through a "peculiarly limited, finite" structure involving finite symbol sets and finite numbers of computational steps.

Completed computations. In order for an individual computation to have taken place, a determinate final state must be reached. If a determinate final state is not reached then the result obtained is not a symbolic type, and the operation performed is not a transition between one symbolic type and another. The functionality of computation is not achieved unless the transition is actually executed. Note that this definition of computation is much closer to the way in which people actually use computing devices than platonic conceptions of computations involving potentially infinite tapes and machines which (potentially) never halt.

Finiteness of computational length. As a consequence of the completion requirement, a computation can only have a finite number of computational steps if it is to yield results in a biologically or humanly relevant time period.

Closure. Computations necessarily involve transitions within sets of states that have already been distinguished, that have been predefined. If there were a state transition into an undefined or indeterminate state or the failure of a state transition to produce a result within a relevant time period, then the functionality of computation would be lost. For this reason, computations involve finite sets of symbolic states, and all operations must stay within those finite sets of states.

The concept of computation here thus corresponds to Ashby's definition of a determinate machine (Ashby, 1956, p.24) or state-determined system (Ashby, 1952, p.26), Minsky's definition of a finite state machine (Minsky, 1967, p.11), or a Turing machine with a finite tape which halts within a finite number of steps. It also corresponds to a strict finitist-constructivist position with respect to (uninterpreted) formal systems (see chapter 7 and appendix 1).

Measurement

A computation is a process in which the final state is contingent only upon the initial state. A measurement is a process in which the final state of the measurement is not completely determined by the initial state; that is, that the transition is not solely due to the type property of the initial state.

Measurement converts undefined, unspecified (in Bohm's terms, implicate) interactions into a finite number of definable, specifiable, explicit categories which are interpreted as symbolic entities by some other system. It is the process of constructing categories (see Pattee, 1985 for an overview), of partitioning the world of possible interactions, of constructing a *semantic* relation between the world and the symbols the measurement produces.

Alternative discrete outcomes. The functionality of measurement necessitates that more than one outcome be observed over the course of observing that organism or device. For each discrete outcome of each measurement there must be a corresponding distinguishable symbol vehicle or token which represents that outcome. As a result, the number of distinguishable tokens or symbolic states must be as large as the number of possible measurement outcomes.

These tokens, which represent discrete outcomes, must be treated as symbols in the subsequent behavior of the organism or device. Thus there must be a symbolic operation which follows the measurement which has symbolic inputs, either a computation or a control.

If only one outcome is ever observed, then the observed behavior of the physical system will be describable in terms of type transitions, since all inputs always yield the same output.

In this case no information flows from the environment to the observer, and the observer gains no knowledge of interactions beyond the measuring device. Measurement is useful to an organism or device for the same reasons perception and cognition are useful to organisms: the behavior of the organism can be made contingent upon events happening in the organism's environment. If the measurement ceases to be contingent upon these external events, then its usefulness to the organism is completely eliminated.

Autonomous behavior. Similarly, if the outcome of the measurement is under the control of the organism or device, then no useful information about the environment will be acquired by making the measurement.

The same is true from the observer's perspective: a measurement must not be completely under the observer's control. If the observer has complete control over the experimental set-up, such that the results of the putative measurement are guaranteed, then the results of the measurement are contingent solely upon the actions of the observer and not upon interactions with the environment. When the results are completely specified in this way, the state transitions become deterministic and a type property is consequently implemented. The results of the measurement become functions of the symbolic state of the organism or device. The process becomes a computation.

Measuring devices. Measurement entails the construction of a measuring device, the setting of that device to a reference state (making the measurement replicable), and allowing the device to interact with the world. In the last chapter the general properties of measuring devices were discussed (Pattee, 1985).

What is being measured? We do not know what aspects of the environment are being measured except by correlating a given measurement with other measurements (precise comparisons can only be made between explicit entities) (Rosen, 1985). The measurement is defined operationally, by the actions of the measuring device. For example, a cell surface receptor can be thought of making a measurement, distinguishing between those substrates to which it binds with high affinity and those which it will not, thus operating as a kind of *switch* (Pattee, 1969). This class of substrates is by no means completely enumerable, and no description may be able to account completely for the behavior of the measuring device. Measurement is not necessarily a representation of something else, only an interaction of a specific sort between the environment and the measuring device.

Selection

Selection is a subspecies of measurement, a measure of performance which affects the structure of the organism's responses rather than as another perceptual input. The functionality of selection is to interact with the world and to gauge how well the organism/device accomplishes a particular task or goal and to produce a symbolic output based upon the organism/device's performance. *As such, selection implements a pragmatic relation rather than a purely semantic one*. The symbolic output is then used to specify the structure of the next generation or iteration of the organism or device.

In biological evolution, the phenotypes of a population of organisms interact with their environments, and those which survive to produce the most offspring propagate their particular genetic traits into the next generation. Thus the differential survival and fecundity of different phenotypes, selection, changes the symbolic composition of the genomes in the resulting population. Different sequences of DNA are inherited by the next generation depending upon the performance of the respective parent organisms.

Selection is, generally speaking, not a well-defined property. In the case of biological evolution, the specific factors which determine a given organism's survival are generally unknown. Like other measurements, what is being measured may not be well defined except through the actions of the measuring device itself. In many situations the only way to assess an organism's survival ability in a given environment is to introduce it into that ecological context. Countless minor ecological catastrophes have occurred through the unforeseen effects of introducing organisms (e.g. kudzu) into new environments, precisely because the selective process was not well-characterized for those contexts.

In the case of trainable machines such as a perceptrons, neural nets, or genetic classifiers, selection plays the role of evaluating the real world performance of the device, causing changes in the input-output function the device is performing.

Control

Control is the process by which explicit, symbolic inputs become converted into nonsymbolic interactions with the environment. Control, like measurement, implements a semantic relation. Once the control device (effector) is activated by some symbol, its actions are nonsymbolic in character, whose ramifications in the world cannot be completely described symbolically. (If they could be completely described and predicted, then they would appear to us as part of our formal description, and the entire process would become a computation.) A control process is one in which the input, but not the output, can be formally, deterministically described. The effectors of most robots are such control devices. They take symbolic inputs and convert them into nonsymbolic actions. While their inputs can be described completely, in terms of type properties, the effects of their outputs cannot be so described; at some level there are always unforeseen and unforeseeable effects stemming from those aspects of the world which are unrepresented in the symbol systems which control them.

Construction

Construction is a subspecies of control. While control processes are thought of as mainly affecting the environment, construction is a control process whose observable effects are primarily on the organism or device itself. An example of a construction process would be the symbolic direction by DNA for the synthesis of proteins, which fold up in a nonsymbolic interaction into the structural proteins and catalytic enzymes of the cell.

In terms of analog and digital processes, construction would be a "mixed digital-analog" process. Von Neumann's kinematic model of a self-reproducing automaton involved such a mixed digital-analog constructor arm which consisted of motor units driven by neurons. In this case the symbolic description of the automaton (in the logical organs, neurons) directs the nonsymbolic, kinematic motions of the constructor arm.

Construction and control are not completely separable processes in their ramifications, since nonsymbolic interactions have effects on both the organism/device and the environment. If a distinguishable change in the device's behavior can be seen, then the process can be considered as a construction. In addition, the environment can influence the constructive process by mutating the symbolic part or modulating the "phenotypic" expression of the symbolic part. Further, control processes can indirectly influence construction by in effect choosing the environment in which construction takes place (Waddington, 1959; Oyama, 1985).

Nonsymbolic interaction

Nonsymbolic interaction is physical interaction for which there is no available symbolic description of either inputs or outputs. Neither inputs nor outputs are encoded, symbolic entities. Nonsymbolic interaction encompasses action completely unmediated by symbol creations or transformations. It must be measured, encoded in order to be described. Nonsymbolic interaction has the same status as

continuity in mathematics: we have discrete symbols representing entities that we interpret as continuous, but no actual continuity in the formalism itself. Analog computers and the compass and straightedge constructions of high school geometry come close to continuous representation, but these are not formal, completely replicable processes. Continuity can only be talked about precisely through discrete symbol strings. Even in the case of so-called fuzzy sets, the formalism used to describe them must always be crisp. Likewise, nonsymbolic interaction can only be represented in formal systems which do not themselves employ the process.

Nonsymbolic interactions form the basis for many nonsymbolic, purely analog devices we use in everyday life, like ball-and-valve centrifugal governors and manual can openers, as well as the substrate for analog computers. While purely analog, nonsymbolic structures and interactions can be adaptive and evolutionary in the senses to be developed below, because they are not discrete, their characterization cannot be precise and consequently, it is difficult to agree upon when a new or different function has arisen. For this reason, nonsymbolic interaction will not be incorporated into the devices considered in chapters 6-11, because we are concerned with the interrelationships of the symbols in our devices and their relations to the nonsymbolic world. Instead, whatever nonsymbolic interactions there are will appear as outside the symbol-relations of the device and in the environment. It should also be remembered that symbols ultimately only arise out of these nonsymbolic, law-governed interactions.

Observable correlates of symbolic-nonsymbolic transformations

The task of distinguishing between symbolic and nonsymbolic interactions and between the processes of computation, measurement, control, and nonsymbolic interaction is a formidable one. From the point of view of an observer looking at some physical system, how does the observer recognize that a measurement, computation, or control has occurred in that system? While a fuller, more rigorous treatment is beyond the scope of this work, this section will give a rough outline of how this question could be answered unambiguously given a particular observational frame. These ideas are still under development, testing and revision.

This question is relevant to the taxonomy of device types presented in chapters 6-10. The behavioral limitations and capabilities of a given device depend upon its structure-function relationships, i.e., which functionalities are implemented and how they are connected together. If the various functionalities cannot be clearly distinguished, then the taxonomy is not a rigorous one, and the taxonomic categories are not distinct from each other. The taxonomy would not capture the essential differences that make open-ended category-constructing devices qualitatively different from other ones already in existence.

One of the difficulties in articulating this question is that the observer's states themselves are results of measurements, so the explanation involves processes that it attempts to explain. The same is true of computation. How does one know a computation has occurred (without error) unless one can oneself carry out a computation? What we want is an account which is consistent for both the observer and an observer being observed. There would then be reciprocity between the two observers' epistemic positions and a self-consistent relativity would be achieved. (This is similar to the problem of other minds in the mind-body problem.) For the moment we will lay these questions aside and assume for the present that the observer has at the very least all the functionalities we would want to attribute to any other organism or device.

We will start with an observer with a given set of measuring devices interacting with a physical system, which could be an organism or a device. The observer is free to choose alternative measuring devices and the degree of resolution associated with each device, but once the observational frame is fixed, then all classifications are made in reference to that fixed frame. The states of the observer's model will be determined by the measuring devices s/he chooses. These states encompass all the distinctions the observer can make on the physical system. If the observer so chooses, s/he can eliminate individual states by changing the frame to eliminate the distinction that constitutes the state. In addition

the observer has the physical means of carrying out computations, as well as the means of constructing, calibrating, and preparing the observational frame.

It should be kept in mind that any observational frame is finite and limited, and as such constitutes a drastically impoverished image of the indefinitely manifold interactions that are taking place in the real physical system. Similarly, the observational criteria for labelling a given state-transition as either a "computation," "measurement," "control," or "nonsymbolic interaction" are not to be construed as the full meaning of these concepts within the broader theory.

Consider the set of observed states and state transitions shown in figure 5.4. We will assume that this behavior has been observed repeatedly. Dotted lines indicate that the state transition shown *only sometimes* occurs; solid lines indicate that the state transition shown *always* occurs over the observational period.

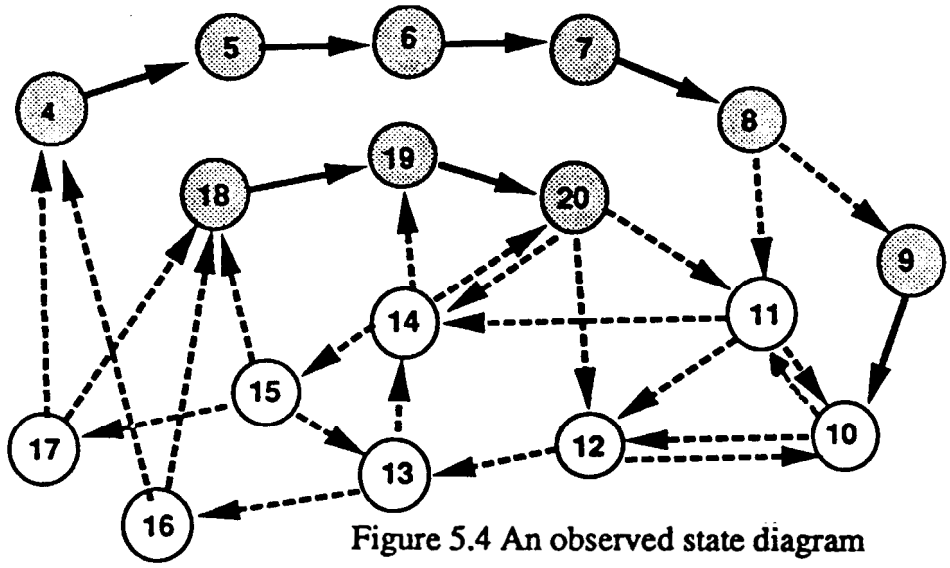

Figure 5.4 An observed state diagram

In the diagram each state is labelled either "symbolic" (shaded) or "nonsymbolic" (unshaded) according to the kinds of relations it has to its predecessor and successor states. A state is labelled "symbolic" if it has one successor state or if all of its predecessor states are already labelled symbolic states. All other states are labelled nonsymbolic states. The entire diagram can be labelled this way, beginning with the labelling of all those states and proceeding to those states which are symbolic by virtue of their predecessor states.

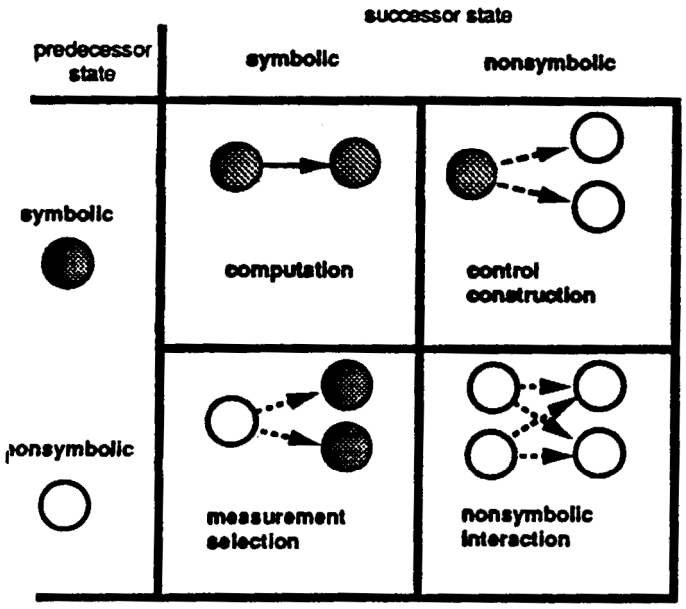

Figure 5.5 Observerd state-transition correlates of the primitive symbolic-nonsymbolic transformations

The observed state-transition correlates of the symbol-matter transformations of figure 5.3 are given in figure 5.5. We will go through each process individually. The diagram should be interpreted as follows. The predecessor state of each transformation is the initial or reference state into which one brings the system in order to carry out the transformation, the desired state transitions. For example, if we want to perform a measurement, we must first maneuver the measuring device and the experimental context into a "prepared" reference state, and then allow the physical system to run its course. If we wish to perform a computation, we maneuver the computing device (by means of programs and data) into the desired initial state and let the physics take over. We let the machine run once we have set up the requisite conditions.

Recognizing when a computation has occurred. A computation as we have said is a transition from symbol to symbol based completely on the property of type and describable by deterministic rules. This implies deterministic observed state transitions. From the observer's point of view a computation will have occurred when some state is *always* followed by the same successor state *over the period of observation*. This will be called a deterministic state transition. A sequence of deterministic state transitions will be a computation from the observer's point of view, since concatenations of computations are computations.

In figure 5.4 the state transitions of 4-to-5, 5-to-6, 6-to-7, 7-to-8, 18-to-19, 19-to-20 and 9-to-10 are computations since they are deterministic state transitions. Note that in the sequence 4-to-8, the transitions 4-to-6, 5-to-7, 6-to-8, as well as 4-to-7, 5-to-8, and 4-to-8 could be labelled as computations if the observer changes the observational frame to drop the intermediate states. An analogous reduction of 18-to-20 could also be done.

While there is no guarantee that the observer's states are recognizing *exactly* the same type properties as the organism or device, the observer's states are close enough so as to completely capture the behavior of the physical system over the relevant observational period. Note also that the physical processes which we so identify as computations are also those which could be used by us to carry out formal operations. The symbolic states of our observational frame become the primitive distinctions (or distinguishable tokens) of our formal system and the computational state transitions become the formal operations on those distinctions.

This operational definition of computation is therefore similar to a observer-centered, phenomenal formulation of the causality principle in physics. Computations are those observed processes for which the observer can invoke the causality principle.

A universal formula without an individual knowing it is a vague phrase. Let us therefore modify Laplace's proposal by substituting man in place of the demon. The causality principle is then valid if it is possible for the scientist, on the basis of known laws, to reconstruct the past and to project into the future when the present state of the world is known, or a portion of it, is completely known.

Postponing the shortcomings of this definition as stated, we cannot fail to see that it enjoys the definiteness and directness which are typical of all positivistic pronouncements: it can be decided at once whether it is applicable or inapplicable to the world as the scientist knows it. (Lindsay & Margenau, 1936, p.519)

For Lindsay and Margenau the relative, observer-centered perspective and its limited explanatory domain are its shortcomings. Clearly a classical theory which is empirically consistent is preferable if it can be constructed for a given domain, but there are very few domains outside of classical physics which have been brought under a comprehensive, predictive theory. For our purposes, this perspective, with its "definiteness and directness" is an antidote to the vagueness which surrounds most contemporary discussions of computation in physical systems.

Recognizing when a measurement has occurred. A measurement involves a nonsymbolic interaction potentially producing two or more a symbolic outputs. The nonsymbolic interaction is not based upon type, so the state transition as a result of such a interaction is a nondeterministic one from the observer's point of view. The output states of the process are symbolic, i.e. each output state is involved in a subsequent deterministic state transition. Thus, a measurement process would involve a nondeterministic transitions followed immediately by a deterministic transition in each of the pathways.

A measurement occurs when the predecessor state is contingent, when the predecessor state has two or more potential (observed) symbolic successor states, and when the transit from the nonsymbolic predecessor state to the symbolic successor state occurs.

A predecessor state can have in addition to its symbolic successor states any number of nonsymbolic successor states, but transits to these states are not measurements because the result of the transit is not interpreted symbolically.

Recognizing when a control has occurred. A control involves a symbolic input and a nonsymbolic output. When a deterministic state transition is followed by a nondeterministic one, then a control process is identified.

In figure 5.4 a control has occurred when a symbolic state (by virtue of its symbolic predecessor states) with multiple successor states transits to one of its next states. Thus the transitions 8-to-(9 or 11) and 20-to- (11 or 12) would be classified as control transitions.

Changing the frame. Frames can be changed by adding new distinctions or dropping existing ones. New distinctions require new measurements and measuring devices, which can either thought of either as new observables or finer resolution of existing observables. The two cases are observationally equivalent, the effect being to give the observer more distinguishable states. Dropping existing states is much easier since it involves ignoring distinctions which were made previously and consequently can be accomplished by redrawing and relabelling pre-existing state transition diagrams to combine those states which were separated by the distinction. It should not be forgotten that dropping states is as much of a change in observational frame as adding new ones.

It may be in a given observational frame a given state transition will be found to be dependent upon more than one previous state. In figure 5.4 states 4 and 18 both have predecessor states of 16 and 17. In this case the observer can combine previous states (16 and 17) to form new observable states (e.g. the state of 26-or-17, the union of the two) that allow for the causal relationship to be easily apprehended. The new frame will thus have different state definitions and hence different functional decompositions.

In figure 5.6 a series of frame changes is shown where distinctions are dropped and states are recombined. The classification of the processes involved changes from completely nonsymbolic

interactions to a measurement and eight computations to a measurement and two computations depending upon how which distinctions are dropped.

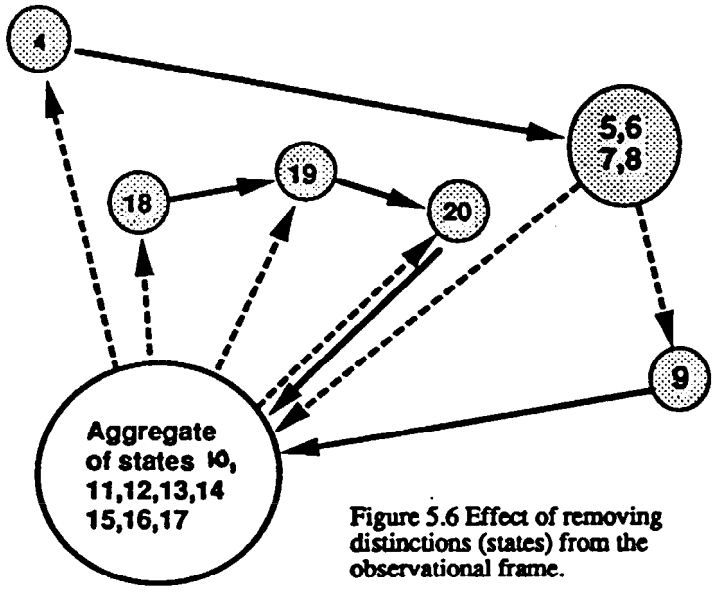

Figure 5.6 Effect of removing distinctions (states) from the observational frame.

In figure 5.7 the effect of adding two new observable distinctions and their accompanying states is shown. The observed behavior of the original system is shown in the first diagram, and that of the augmented system is shown in the second. Three nonsymbolic interaction state transitions followed by two computations become a measurement followed by six computations when two states are added to "resolve" some of the nondeterministic state transitions. Note that the boundaries of the syntactic, "rule-governed" computational part are extended outwards when the two states are resolved, and the contingent, semantic interface between the symbols of the computational part and the nonsymbolic interactions changes as a result. If we had added states within the computational part, say intermediate between states 5 and 6, and these all had deterministic state transitions, the semantic relations would be unchanged.

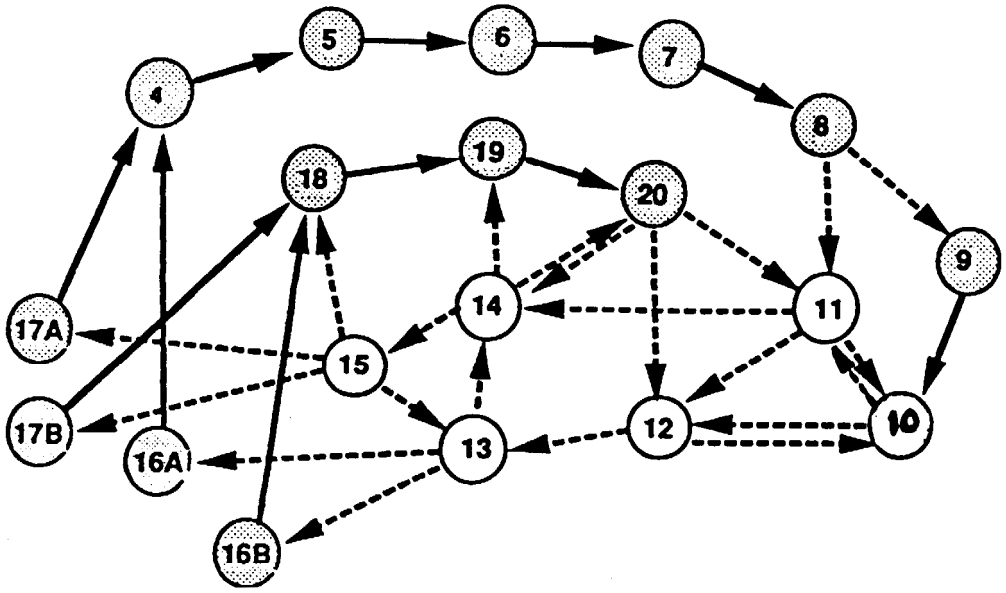

Figure 5.7 Effect of adding two new distinctions (states) to an observed state diagram. In the process former states 16 and 17 were resolved into 16A, 16B, 17A and 17B, resulting in four deterministic state transitions where there were four non-deterministic ones. Here the boundary of the computational part has been moved outwards.

On the irreducibility of the primitive transformations

The rough image of the last figure, in being able to "resolve" all indeterministic state transitions, of course, was the idea behind "hidden-variable" theories of quantum mechanics (Jammer, 1974; Bohm, 1952; Bohm, 1957). A claim based on such an image is somewhat gratuitous unless one can actually find the new observable states which resolve the indeterminacies, which would involve new measurements and measuring devices. In quantum mechanics no new observables corresponding to the sought after hidden variables were ever found to bring the theory to complete prediction of individual state transitions. Bohm showed that any such hidden variables would have to have magnitudes on the order of the uncertainties associated with the measurement process and von Neumann produced a proof which showed that no conceivable set of hidden states could resolve the observed indeterminacies (Bohm, 1957, pp.85-86). There have been further disputes since, but these have not challenged von Neumann's main conclusion (Wheeler & Zurek, 1983).

The operations of measurement, computation, control, and nonsymbolic interaction have been here defined so that, within a given observational frame, they appear as mutually exclusive operations. Care has been taken to ground the definitions in observable distinctions so that their meaning has more than metaphorical significance.

In part this comes out of controversies surrounding the status of measurements and the mathematical formalisms involved in quantum mechanics, the "measurement problem" alluded to in the last chapter. Bohr's interpretation of the problem was that measurement was itself a irreducibly primitive operation, that it could not be included in the formal part of the model.

Von Neumann (1954) showed that the attempt to include the measuring apparatus in the formal part of a physical model leads to infinite regresses: the state of the apparatus must then be predicted using yet more measurements to establish the initial conditions of the physical system *cum* measuring device. A similar argument involves the irreducibility of the construction of the physical apparatus. Clearly the apparatus must come into being: it must be constructed and calibrated. To include the construction and calibration in the formal part would necessitate measuring the initial conditions of the parts of the

apparatus, which would require measurements, necessitating construction of the appropriate measuring devices. And so on.

The essential distinctions on which these arguments turn are that initial conditions in the formal part are independent of the rules which govern the computations of the model. These computations, of course, represent the physical laws believed to be operating in the real system.

The major implication is that at least three primitive functionalities are needed to construct formal models of the world, that the functionalities of measurement and control cannot be completely subsumed into the computational part. This contradicts the currently popular idea that "everything is simulatable," which is Church's Thesis in its strong form (see Rosen, 1962, 1986; Pattee, 1988, 1989; Conrad, 1983; Hofstadter, 1979; for discussions). From this point of view, computations (nontrivially defined) can never completely take the place of either controls or measurements, so there are functionalities which are not computable, and Church's Thesis in its strong form is false or at best a badly posed question. While computations can *represent* measurement and control processes, computations cannot *implement* them (cf. Rosen, 1986. One cannot achieve semantic operations by purely syntactic ones, and vice versa. Computations cannot avoid measurements (Pattee, 1988).

Chapter 6 Adaptivity

> *Plasticity*, then, in the wide sense of the word, means the possession of a structure weak enough to yield to an influence, but strong enough not to yield all at once. Each relatively stable phase of equilibrium in such a structure is marked by what we may call a new set of habits. Organic matter, especially nervous tissue, seems endowed with a very extraordinary degree of plasticity of this sort; so that we may without hesitation lay down as our first proposition the following: that *the phenomena of habit in living beings are due to the plasticity of the organic materials of which their bodies are composed*. (James, 1892, p.2)

Three levels of adaptivity

William James' concept of plasticity when applied to our concepts of measurement, computation, and control arenas allows for the construction of devices which automatically solve three basic types of problems. These are problems of *classification*, problems of *decision*, and problems of *effective action*. Alternately, we could call them problems of encoding, code manipulation, and decoding.

Finding effective semantics. The problems of effective classification and of effective action are semantic in character. They are related to the finding and implementing useful relations between symbols and the world. Finding the appropriate variables to effectively model a given physical system or finding the appropriate features needed to effectively encode human handwriting are examples of classification problems. These problems involve the optimization of measurements which are made; consequently these problems are semantic in nature. Finding the appropriate means of milling a piece of metal or of shaping the contours of an automobile to reduce aerodynamic resistance are problems of effective action. These problems involve the optimization of controls, and hence are also fundamentally semantic in character.

Finding effective syntaxes. On the other hand, problems of decision and coordination involve choosing between discrete alternatives given a defined situation. They involve finding useful mappings of symbols to symbols, and are therefore completely syntactic in nature. Thus, we can talk of problems of a semantic nature versus those of a syntactic one. Adaptive mechanisms involving either syntactic or semantic functionalities are capable of solving their respective types of problems. A taxonomy of adaptivity can be constructed along these distinctions, between devices according to whether they are adaptive in their syntaxes and semantics.

Implementing effective solutions. Once the problems of finding the semantics and syntactics needed to solve a problem, there remains the task of implementing the problem solution in a reliable and efficient way. Once the semantic and syntactic search problems have been solved there is no longer a need for adaptivity in those domains; what we want at that point is the most economical means of deterministically solving the problem at hand. Completely nonadaptive, completely reliable, "formal" devices are desired.

The taxonomy. Three broad categories arise: completely nonadaptive devices, devices with adaptive syntaxes, and devices with adaptive semantics. These respective types of devices will be called generally "formal," "adaptive," and "evolutionary" (Table 6.1). These can also be seen as stages in problem-solving, with evolutionary devices being the search for effective semantics, adaptive devices being the search for effective syntactics, and formal devices the reliable, efficient implementation of both. Each type of device automates a step in the problem-solving process.

device type	plasticity	syntax	semantics
formal-computational	fixed syntax	performance-independent	No inherent semantics
formal-robotic	fixed syntax fixed semantics	performance-independent	Fixed sensors and effectors
adaptive	adaptive syntax fixed semantics	performance-contingent	Fixed sensors and effectors
general evolutionary	adaptive syntax adaptive semantics	performance-contingent	performance-contingent syntax performance-contingent semantics

Figure 1.5 A taxonomy of types of adaptivity.

The choice of these most general labels reflects their predominant usage and connotations. "Formal" implies well-defined, replicable, perfectly reliable symbolic behavior. "Adaptive," in biological terms generally connotes a change in internal structure within the framework of fixed limitations. It can refer to either physiological adaptation (e.g. growing fur in the winter) in which the limitations are fixed physiological mechanisms or to microevolutionary adaptation (e.g. the sudden predominance of dark colored moths during London's industrial revolution) in which the limitations are the range of allelic alternatives in the current gene pool. "Evolutionary" has more morphological, macro-evolutionary connotations involving qualitative changes in body structures and accompanying emergence of new functions. Rather than selecting from pre-existing alternatives, "evolutionary" conjures up images of the creation of new alternatives.

Devices can be both "adaptive" and "evolutionary," having both adaptive syntaxes and adaptive semantics. These devices will be termed "general evolutionary" devices. Two types of formal devices will be also distinguished. Those devices which are purely syntactic in operation and have no inherent relation to the world at large except through human intervention will be called "formal-computational" devices. These would include computers whose inputs and outputs are completely symbolic as well as purely mathematical operations. In and of themselves such devices have no inherent real world semantics. The second class of devices, "formal-robotic" devices, will include formal devices which are connected directly to the world via built in sensors and effectors. These would include pre-programmed robots as well as fixed formal models with their measuring apparata. These devices have inherent, if static, real world semantics (figure 6.2):

adaptivity class	biological analogy	device examples
formal-computational robotic	Fixed genetic programs Fixed physiological & neurological S-R patterns	Formal systems, computer programs & simulations Fixed algorithms Fixed sensors, effectors
adaptive	Microevolution: Selection of optimal combinations of pre-existing traits	Perceptrons, trainable classifiers, genetic algorithms, trainable control programs
evolutionary	Macroevolution: Creation of new species New structures/functions/traits	Immune system Structural evolution Robots w. self-constructed sensors, effectors

Figure 6.2 Biological and artificial examples of types of adaptivity

The general idea of degrees of adaptivity is embedded in the biological distinctions between physiological, microevolutionary, and macroevolutionary changes. Implicit in these levels are notions of fixed, "genetically programmed" biological mechanisms, "variable-parameter" genetically-modulated mechanisms, and the evolution of qualitatively new mechanisms. Here levels of adaptivity are cast in terms of the plasticity of the genome and the phenotypes it produces.

Correspondences with other taxonomies

In order to facilitate understanding, the taxonomy of adaptive levels outlined above will be compared to other existing classifications, noting similarities and differences.

Piaget's knowledge types. A rough correspondence can be made to Piaget's division of knowledge into logico-mathematical knowledge, experience-acquired knowledge, and evolutionarily acquired knowledge if one does not adopt its ontogenetic/phylogenetic separation literally. Piaget's divisions can be taken roughly as "experience independent knowledge" (or the a priori knowledge of Kant), "experience-dependent knowledge" (or empirical data), and structural or genetic knowledge (implementing the empirical categories).

> From the point of view of biological problems that they raise, we can distinguish three forms of knowledge that result from the exercise of the cognitive functions in man-at least when he reaches a certain level of civilization. In the first place, there is the vast category of knowledge acquired by means of physical experience of every type, that is, the experience of external objects or of whatever appertains to them, abstraction being made of objects as such. At once it will be seen that this means an infinite extension of learning behavior or practical intelligence, but there are all sorts of novel aspects to it which need explanation. In the second place, there is the extremely restricted category-in fact, it is debatable whether it has any real extension-of knowledge structured by hereditary programming, such as may be the case with certain perceptual structures (seeing colors in two or three dimensions in space). The limited nature of this second category at once raises a great biological problem, just because of the contrast between it and the great variety of instincts in animals. In the third place, there is the category of logico-mathematical knowledge, and this is at least extensive as the first. Such knowledge achieves independence of experience, and, even at the stage where it is still bound up with experience, it seems to spring not from objects as such but from the general coordinations of the actions exerted by the subject on the objects around it. (Piaget, 1971, pp. 226-267)

In contrast with Piaget's assessment of genetic, structural knowledge as an "extremely restricted category," it will be argued that the genetic, structural knowledge is necessary for acquiring empirical knowledge and assigning meaning to logico-mathematical coordinations. In model-theoretic terms, it is at least as important to select the appropriate variables among an indefinitely large number of alternatives, as it is to gather data about those variables or to formulate a formal model involving those variables.

Ashby's taxonomy of adaptivity. Those familiar with Ashby's taxonomy of adaptive systems (Ashby, 1952; Moray, 1982), which distinguishes between state-determined systems, Markov systems, and self-organizing systems, will find a number of correspondences between his taxonomy and the one presented here. The behavior of formal-computational devices would appear as state determined systems and the behavior of evolutionary devices would have behavior associated with his self-organizing systems, where variables themselves are free to change. The principal divergence lies between the behavior of flexibly-deterministic syntactically-adaptive devices vs. his stochastic Markov systems.

Gause and Rogers taxonomy of adaptivity. The idea for the device adaptivity taxonomy presented here comes from the machine learning taxonomy of Gause and Rogers (Gause & Rogers, 1982).

Their classification system consists of four classes, based upon whether a given system is adaptive or nonadaptive and whether its performance level is low or high ("stupid" or "intelligent"). An *adaptive system* changes its behavior in response to its performance in an external environment. A *non-adaptive*

system maintains a fixed behavior regardless of its performance. In control-theoretic terms, adaptive systems are closed-loop devices, while non-adaptive systems are open-loop devices.

A non-intelligent, nonadaptive device (e.g. a poorly designed computer program) is an *inane system* while an intelligent, nonadaptive device (e.g. a well-designed expert system) is a *meretricious system*. A non-intelligent, adaptive device (e.g. Ashby's homeostat) is a *homeostatic system,* while an intelligent, adaptive device (e.g. neural nets) is a *learning system*. The homeostatic system clearly takes Ashby's homeostat as its archetype. The purpose of the homeostat is not to learn to improve some external function, but to adaptively maintain stability in the face of environmental perturbations.

These general device types are clearly coupled with their respective characteristic behaviors. Plotted relative to time, nonadaptive systems have constant performance levels, due to their fixed internal structure, while adaptive systems have changing performance levels. The performance of a learning system generally increases over time, plateauing when maximum possible performance given the features and actions used is achieved. The overall performance of the homeostatic system fluctuates around a constant level.

Barto and Sutton. Barto and Sutton (1981) surveyed the field of adaptive devices, using the cybernetic and control-theoretic terminology of open-loop vs. closed-loop functioning. Here open-loop devices are those devices whose behavior is not affected by their outputs. Open-loop devices correspond to formal devices, as defined here. Closed loop devices correspond to adaptive devices. In their survey there were apparently no closed-loop devices which altered their own input or output transducers, so there were no devices corresponding to the evolutionary devices of this paper.

C. West Churchman's Inquiring Systems. Churchman (1971) has connected various seminal philosophies with cognitive learning strategies. In the taxonomy here formal devices are Churchman's Leibnizian inquirers; adaptive devices are equivalent to his Lockean, associative inquirers, while evolutionary devices can be roughly seen as Singerian inquirers.

The role of human beings in the taxonomy

It is common in computer science and artificial intelligence to tacitly include human programmers and users in their conception of the devices themselves.

Separation of devices from designers. For the sake of clarity we will analyze the various devices just as we would analyze the capacities and limitations of any organism, independent of any human-device cooperation. This will make it possible to see more clearly what parts of the devices are adaptive and autonomous of their designer-observers and which parts are not. It also more directly addresses the problem we are most interested in, namely, how we design devices which *by themselves* construct their own semantic and syntactic primitives.

The necessity for separating the devices from their human designers becomes readily apparent if we consider the consequences for the taxonomy for not making the separation. Because of the extreme plasticity and creativity of human beings, relative levels and types of adaptivity would be difficult, if not impossible to distinguish. In our taxonomy, a human-plus-computer combination could be a formal-computational device if the human only inputted and outputted symbols without interpretation, a formal-robotic device if the human maintained a fixed interpretation of the computer's actions and acted accordingly, an adaptive device if the human being reprograms the computer to optimize its performance, and an evolutionary device if the human being decides to reinterpret the meanings of the symbols manipulated by the computer.

Chapter 7: Formal-computational devices

> A "machine" is that which behaves in a machine-like way, namely, that its internal state and the state of its surroundings, defines uniquely the next state it will go to. (Ashby, 1962, p.111)

Since the days of Babbage's Analytical and Difference Engines, the capacities and limitations of computing devices have been hotly disputed. With the advent of electronic computers and artificial intelligence, this debate has intensified, but without the benefit of any physically rigorous definition of what a computation is, i.e. *how we would know one if we saw one in nature*.

This chapter attempts to give a rigorous definition of what here are called *formal-computational devices*. These devices are very close to the physical devices we know as digital computers, but bear some important differences from the idealized computing device of Turing (1936). The main distinction is that here we require our devices to be physically realizable and hence finite in their storage capacities. The resulting concept of a formal-computational device is close to that of a physically implemented Turing machine having a finite tape. Because the Turing definition is so deeply embedded in computer science and artificial intelligence that it has effectively ceased to be questioned, much of the chapter (and appendix 1) are devoted to presenting the rationale for the alternative, physical conception of computing devices.

The second half of the chapter examines the capacities and limitations of formal-computational devices *with respect to the problem of automatically generating new external semantic relations*. The enormous capabilities of digital computers are well-understood now that these devices are widely in use throughout our society. In contrast, their limitations are rather poorly understood, and it is for this reason that limitations are much more analyzed here than capacities. It should be made clear, however, that nothing said here is in any way meant to disparage the use of computational devices, which have become indispensable parts of our scientific analytic armentorium.

Definition of formal-computational devices

Given a particular, fixed observational frame, a *formal-computational device* is one whose observed state transition behavior over some specified observational period can be described completely in terms of computations. To describe a given observed behavior as a computation (see chap. 5), it is necessary that 1) a fixed observational frame be specified, such that it can be communicated to another observer who can replicate the observations 2) for each observed state there be no more than one immediate successor state, such that a deterministic observed state transition rule can be constructed for all states, and 3) a final state is reached within some finite, relevant length of time.

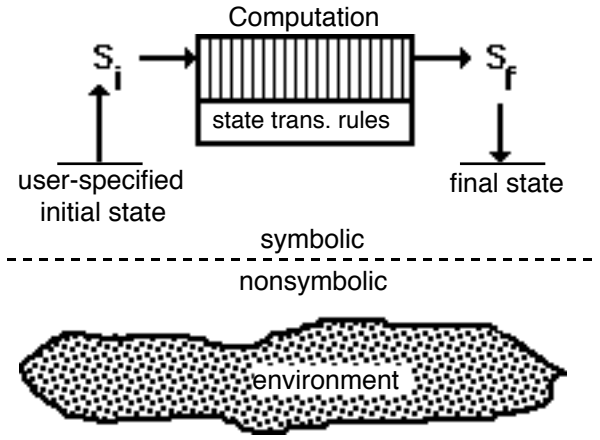

Figure 7.1. Structure-function relations for a formal-computational device. Note that the device operates completely within the symbolic realm and that the environment does not affect the structure of the device.

The structure-function relations of a formal-computational device are illustrated in figure 7.1. The input-output behavior of the device is *completely independent of events in its environment* once the inputs have been given. An observed state diagram of the device itself is given in figure 7.2. The observed state transitions corresponding to the device must all be computations. If we were also to include environmental states, the observed state transitions would appear as shown in figure 7.2. Note that, by definition, there cannot be connections between the environmental states and the device states without there being measurements and/or controls involved. The inclusion of measurement and controls in the device itself would make the device appear as a *formal-robotic* one, to be discussed in the next chapter.

Thus stated the definition of computation corresponds to a physically-based strict finitist-formalist conception of an uninterpreted formal procedure (see appendix 1). Formal-computational devices implement formal procedures.

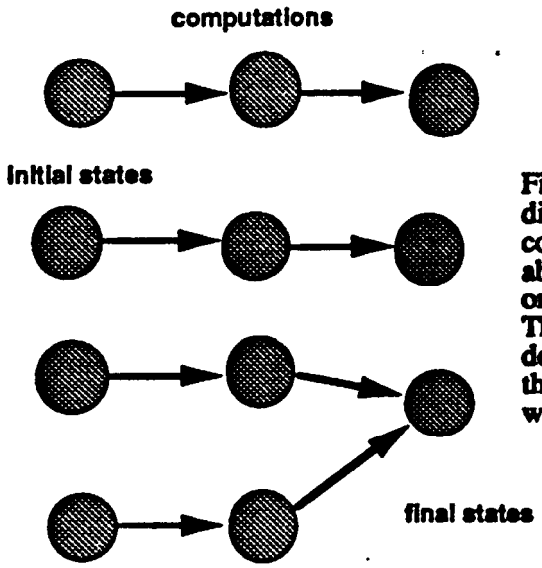

Figure 7.2 Observed state transition diagram for a very simple formal-computational device. Note the absence of any measurement, control, or nonsymbolic interaction transitions. The initial states of the computational device are prepared by the user, so they are intentional starting points, without any antecedent observed states.

The function of formality

The purpose of a formal procedure is to achieve reliability and replicability of result, based upon the unambiguously defined, deterministic physical operations on inherently meaningless discrete physical tokens, such that all observers of the procedure can agree precisely upon the correct outcome of an operation given an agreed upon initial state. "The manipulation of sequences of symbols, according to definite syntactical rules, is the essence of formalization." (Rosen, 1987, p.1). Formal procedures are completely syntactic processes; they can be described completely in terms of rule-governed transformations of symbolic types. The symbols which are manipulated in formal procedures have no necessary, inherent semantics; they are not semantic entities. The symbols of a formal system can be related to the world at large in any number of ways, by means of external linkages in the form of human interpreters, biological organisms or through measuring and control devices.

The concept of a formal procedure is close to the intuitive notion of an effective procedure:

> In programming digital computers we are always concerned with such procedures--with exhibiting concrete results that can always be expressed in numeric form. These procedures lead to results in an unequivocal fashion. If they are repeated without change they yield the same results. It is the actual results that are of interest to us; we want to reach the point where we can display them, and we are not satisfied merely by knowledge of their existence. Such procedures are defined by the following characteristics:

1. They are *deterministic* -- implied by the fact that we expect to obtain the same results from the same starting conditions.
2. They are executable in *finite time* and using some *finite facility*. We usually assume, however, that if it becomes necessary to increase this facility during the execution of the procedure, such as by increasing the quantity of paper available for the computation, this will be done....There is a considerable element of unreality in requiring only that our resources be finite without stating some limit on their size. This allows, for example, procedures that might take more time than the reputed age of the universe or that might require storage capacity that exceeds the size of a finite universe....
3. The execution of each such procedure is "mechanical" or "constructive" and can be precisely described so that another intelligence, or perhaps a device, could receive this description and use it to apply the procedure and obtain identical results.
4. These procedures can be *cast in numeric terms*.... They involve objects that can be represented by the natural numbers ... and we can always interpret the operations within such procedures as arithmetic operations....Further, even the statements of these procedures are finite and can, themselves, be represented as natural numbers. (Beckman, 1980, p. 2)

For the most part, this standard definition of formality is the one to be adopted here, but in order for it to be consistent with our definition of computation, potentially infinite processing times, storage capacities, and numerical representations of formal statements must be disallowed. This is due to the "considerable element of unreality" involved in their postulation. A process which generated unreplicable, indefinite, or ambiguous outcomes would be functionally useless as a formal process. A process which failed to produce any result or which took the age of the universe to terminate would similarly be functionally useless to us as a formal process. A brief, but more general discussion of potential and actual infinities in the debate over the foundations of mathematics is presented in Appendix 3. The conception of formality adopted here will be similar to that of the mathematical constructivists, in particular, to that of the strict and "ultra" finitists (Troelstra & van Dalen, 1988, chap. 1).

Physical realizability

Formal *devices* must be physically realizable if formal rules are actually to be executed. The interpretation of this statement is probably the greatest source of disagreement regarding the nature of formal processes, going straight to deep ontological and epistemological issues in the foundations of mathematics.

> We have stated that one approach to making precise the notion of an effective computation is to describe some physical device and to define effective computations in terms of the procedures that can be accomplished on it. (Beckman, 1980, p. 26)

As much as possible, we strive to be concrete. We are discussing *constructible physical devices which we as observers could use to implement computations*. Formal-computational devices encompass the class of physically realizable devices whose observed behavior can be described completely in terms of computations, as defined in the last chapter. As such, these devices all have speed and storage limitations, making them in their appearance equivalent to finite state automata as opposed to Turing machines (as in Minsky, 1967).

> ...finite automata are fixed size, deterministic, discrete, synchronous, finite state machines....All real computers, because of there is always some practical bound on the size of auxiliary storage that can be assumed for them, are examples of finite automata rather than of Turing machines. (Beckman, 1980, p. 256)

Every physically realizable computational machine has a finite set of states. In part this is due to the finiteness of material resources: no tape is truly indefinitely extendible, no stack indefinitely deep. More importantly, the finiteness of the device is related to the finiteness of the observer. Each state of the

device is really an observer-device state. Even if the device could have more configurations than the observer could distinguish, the behavior of the device is only accessible through the observer's states. *Every observer has a finite number of observables and a finite number of distinctions for each observable. Every observer can only distinguish a finite number of states.* This in turn is a consequence of the finite number of distinctions that a measurement process can make.

The finitistic, functional conception of computation along with the necessity for its realization in physical formal-computational devices which can be directly observed is not a trivial semantic quibble or an irrelevant metaphysical distinction. We want to discuss as clearly as possible the capacities and limitations of contemporary digital computers. The finitist account of computations most closely corresponds to how we actually use computer programs and simulations. If a computer program or simulation does not terminate within a humanly relevant amount of time, it's not useful to us, and we redesign it so it does. If a computer program or simulation uses too much storage space, it cannot be executed and we must either redesign the software or find a bigger machine, but, nevertheless, there are limits, even if they are constantly being pushed back. All the symbol manipulations we carry out must be within a "computational envelope" of technological and economic limitations.

Thus, the seemingly more restricted, finitist definition of computation in use here excludes no programs or devices commonly thought to implement digital computations. Importantly, the behavior of all of these finite state program-device combinations can be completely captured by finite state automata descriptions. *Consequently, we do not have to be burdened with issues of computability in this discussion.*

Conditions of observation

> The real world gives the subset of what *is* ; the product space [of possibilities] represents the uncertainty of the *observer*. The product space may therefore change if the observer changes; and two observers may legitimately use different product spaces within which to record the same subset of actual events in some actual thing. The 'constraint' is thus a *relation* between the observer and thing; the properties of any particular constraint will depend upon both the real thing and on *the observer*. It follows that a substantial part of the theory of organization will be concerned with *properties that are not intrinsic to the thing but are relational between observer and thing*. (Ashby, 1962, p. 109)

Whether a given device appears to be implementing computations is dependent upon our means of observing it.

This maxim embodies the observationalist concept of a deterministic system, one in which it is necessary to specify the experimental conditions under which the system appears to be deterministic. This requirement flows from the belief that it is gratuitous to label a physical system as "deterministic" or as "implementing a "computation" without specifying the conditions under which deterministic or computational behavior can be observed directly.

This definition of formal-computational devices is most readily exemplified by existing modern electronic digital computers and other physically realized formal processes, *provided that we select the appropriate means of observing them.*

For example, a electronic digital computer running a program can be seen as a formal-computational device if the observer chooses to measure states corresponding to the two classes of voltages (low and high) which the machine logic gates recognize and respond to. The global machine states of the entire device, including all types of memory, will appear to be deterministic if 1) the machine is operating properly, i.e., without error, and 2) the correspondence between the observer's measured states and the logic gate "states" is close enough.

Similarly, there are many other higher levels in the computer which could be selected which would allow for deterministic description given the appropriate choice of observables. If the computer is running a high level evolutionary simulation, for example, and the observer's states are all program

variables and initial conditions, then each "state" of the simulation will always produce the same successor state, again, if the machine is operating properly.

However, if we observe successive microscopic readings of the voltages on a time scale much faster than the clock speed, such that voltage fluctuations on the microvolt level become our states, then our digital electronic computer will no longer appear to us to be deterministic in its behavior. We could, of course, change the temporal resolution by averaging many fast states together and we could change the voltage resolution by combining microstates into macrostates, but this would be a change in our observational frame, our state definition. By changing to a coarser observational frame, the apparent nature of the device's behavior would change from stochastic to deterministic.

If our observational frame is on the appropriate level of resolution but some relevant aspects of the device's behavior are not observable, then the observed behavior of the device will not appear to be deterministic. An example of this would be if we were observing a cellular automaton simulation in which some of the nodes were hidden. The simulation would appear to be stochastic. Unless we changed our observational frame to encompass all the relevant simulation states, strictly speaking the truncated simulation would not be implementing a computation. Given the states we had at our disposal, we could not reliably always end up with the same results given the same initial state.

Through the frames of gate voltages and simulation states, the physical system would appear to us as a formal-computational device; through the frames of microvoltages and partial simulation states it would not appear to us such that we could describe its observed behavior deterministically.

A similar mode of analysis can be applied to "analog computers." While the internal workings of analog devices are not digital in character, there is always a discrete measurement made when the device is set to its initial state and when the final state is read off. We can take the discrete results of these measurements as our observational states. Because of small perturbations, the input-output behavior of the device may not at first appear to be completely deterministic. If the analog device is stable with respect to small external perturbations or errors in setting the initial state, then it will always be possible to coarsen our observational frame so that the input-output behavior will be deterministic. Always, the criterion is whether the device's observed behavior can be replicated exactly.

For virtually all computer programs, computer simulations, mathematical models and formal procedures we can easily find an observational frame in which their behavior will appear to us as computations. When we choose to observe them in the appropriate way, such that their behavior is deterministic, we classify them as formal-computational devices.

Pseudorandom and chaotic processes

This analysis has bearing on how we regard the symbol strings generated by pseudorandom and chaotic processes implemented on a digital computer. If we view all the relevant global states of the generating process, the symbol sequences will appear to be deterministic, and the device will appear to us as a formal-computational device. In Ashby's (1952) terms, we would have a *state-determined system*. We would be able to completely replicate any particular trajectory of this device by placing the device in a given state and letting it run.

However, if we only view some subset of the global states, the symbol sequences will not appear to us to be deterministic. Instead they will appear unpredictable: random, stochastic, or chaotic. We could not replicate the particular trajectories exactly if we place the system in one of our observed states and let it run, since the state-transitions would be based upon events (the unobserved states) outside our observational frame. Some of our observed states would have multiple successor states.

When viewed this way, the device will no longer appear to us as a formal-computational one; given this truncated set of states, we could not use the device to implement formal procedures, since each state transition would not be deterministic.

Weinberg's *Diachronic Principle* is relevant here: "If a line of behavior crosses itself, then either: 1. the system is not state-determined or 2. we are viewing a projection -- an incomplete view." (Weinberg,1975, p.190) Here our truncated observed system is not state-determined. We would have to add the appropriate states to recognize it as a projection of the system encompassing the global device states, which is a state-determined system.

We will discuss this situation in greater depth when we discuss the issue of emergence in computer simulations in chapter 11.

Programmability

While many physical systems exhibit observed behaviors that are computations, the observer in most instances does not have complete control over what state the physical system is in**.**

A formal-computational device is *completely programmable relative to the computations it can perform* if we as observers have the means to bring the device into any desired observed state. This is related to the discussion of the interpretation of state-transitions in figures 5.4 and 5.5 of chapter 5. The act of programming is similar to the act of preparing the experimental apparatus in a physics experiment, of bringing the physical system into a given state. In the case of computers and computer programs, the computer is initialized, brought to an initial state, then the programs and their data are stored in memory. The initial global state of the machine is the state of all registers and memories. From this initial global state there will be but one trajectory for the device until the device halts in some terminal state.

Which states can be reached by the observer effectively determine which operations can be performed by the user of the device. If only some observed states can be reached, the device is *partially programmable relative to the computations it can perform.* A cellular automaton simulation which only allowed for initialization of all node states with the same values would be such a device, since nonuniform distributions of node states could develop and would be observed, but it would not necessarily be possible to directly specify that the simulation start in a particular state, since there might be configurations which are not be accessible from uniform state distributions.

If no states can be reached, and the device can only be observed, then the device is *nonprogrammable*. Given the appropriate observables the motions of the planets can appear to us as deterministic state transitions, and therefore appear to us as computations. However, we have no means of setting the planets in a particular configuration, so that while we could implement certain very restricted formal operations by waiting for the planets to enter the appropriate state and observing the subsequent transitions, our "planetary system computation device" would not be of much general use to us.

Computational universality

While progammability concerns the accessibility of the observed states of a device, computational universality is related to the range of state-transition behaviors that are available to the device. A device is computation universal if any possible deterministic observed state transition behavior can be implemented with the device. "Thus, a universal machine is one that can produce an arbitrary input-output function; that is, that can produce any dependence of output on input (cf. Newell, 1980, p.147)." A computation universal device can be used to implement any possible function, f: A→B, between a set of observed initial states, A, and a set of observed final states, B, where both A and B are subsets of the set of observed states. Once we have the simplest Boolean functions, all other finite functions can be built up through composition of functions (via the coupling of devices) and recursion (via multiple iterations), limited only by the number of distinguishable states, the capacities and number of state-transition components, and the computational time allowed.

There are several formally equivalent ways we can visualize what happens when we program and run a digital computer.

The first is the total machine state description. Computers can be seen as *very large* directed graphs of global machine states (e.g. Minsky, 1967, p. 24). Contemporary computers are flexible enough so that for all possible functions, there is a set of global machine state transitions which, suitably labelled, will implement the function. In this view to implement a particular function, we bring the device into a particular initial global state which, upon running the device, will lead to the desired final global state. The initial state of the device contains both the program and the data, while the device has its own law-governed physical dynamics. The task of programming is the task of choosing sets of global state transitions which have structures suiting our purposes (i.e. the sets of global state transitions are isomorphic to the functions we want to implement), and labelling them accordingly. This description has the advantage of encapsulating all possible machine states and trajectories in one notation.

The second is to think of the program-hardware as setting up a directed graph of state transitions, while the machine's input data (taken all at once, as a vector) determine its initial state. In this scheme the machine's inputs select which deterministic chain of state-transitions the machine will follow. This scheme has the advantage of separating the state-transition structure of the device from its initial state. Consequently the input-output function of the entire device can be seen most readily from this point of view. This is the description closest to the Hertzian picture of the formal part of the commutation diagram (chap. 4, fig. 4.1).

A third perspective sees the program as setting up a directed graph of computational contingencies, each dependent upon an input. This is the most familiar form of the state-transition diagram for a finite state machine, with machine states as nodes and state-transitions as edges labelled with their corresponding input trigger and output. Given a device which is completely programmable and computation universal we can specify and realize any combination of its initial states and state-transition rules. We can completely specify the structure of the state-transition graph by writing the appropriate program. We can completely specify the path of successive machine states by choosing the appropriate inputs. This description has the advantage of emphasizing the completely specifiable nature of the machine's behavior. It has the disadvantage of spreading the description of a given computational trajectory out in time, making it more difficult to see at one time the entire range of possibilities open to the device.

From the point of view of an external observer, all of these accounts are equivalent in describing the device's behavior, and the account adopted has no effect on that behavior.

> The 'machine with input' or the 'finite automaton' is today defined by a set S of internal states, a set I of input or surrounding states, and a mapping, f say, of the product set I x S into S. Here in my opinion, we have the very essence of the 'machine'; all known types of machine are to be found here; and all interesting deviations from the concept are to be found by the corresponding deviation from the definition.
>
> We now are in a position to say without ambiguity or evasion what we mean by a machine's 'organization'. First we specify which system we are talking about by specifying its states S and its conditions I. If S is the product set, so that $S = P_iT_i$ say, then the parts i are each specified by its set of states T_i . *The 'organization' between these parts is then specified by the mapping f.* Change f and the organization changes. In other words, the possible organizations of between the parts can be set into one-one correspondence with the set of possible mappings of I x S into S. Thus 'organization' and 'mapping' are two ways of looking at the same thing -- the organization being noticed by the observer of the actual system, and the mapping being recorded by the person who represents the behavior in mathematical or other symbolism. (Ashby, 1962, p.111)

Obviously all of these notations rapidly become unwieldy with even modest numbers of states, but nevertheless, they are useful for purposes of discussion. Higher levels of macro-description which reduce to any of these micro-descriptions can be used to simplify the discussion considerably, but these also do not change the state-determined nature of the behavior.

Using any of these accounts, either by choosing the appropriate initial global machine state, or by specifying the state transition properties of the device, *we are able to completely specify completely the behavior of the device*.

> *It is certainly true* that *programming*--the job of specifying the procedure that a computer is to carry out--amounts to determining in advance everything that the computer will do. In this sense, a computer's *program* can serve as *a precise description* of the process the machine will carry out, and in this same sense it is meaningful to say that *anything that can be done by a computer can be precisely described*.* (Minsky, 1967, pp.103-104; * It is important to note that this does not mean that the person who writes a computer program automatically understands all the consequences of what he has done! ...)

All that has been specified will occur and nothing will occur unless it has been specified, as long as the device reliably implements the same state-transition behavior. As far as the programmer's understanding of this specification goes, *even if the programmer cannot foresee every logical consequence of his or her specification, s/he has the means at hand to determine all the logical consequences, i.e. by running the program*.

Formal-computational devices are syntactically static

By definition the formal-computational device must always implement the same input-output behavior during its use as such a device.

> A machine is defined to be a system that has a specific determined behavior as a function of its input. By definition, therefore, it is not possible for a given machine to obtain even *two* different behaviors.... (Newell, 1980, p.148)

Thus the syntax of the device must, by definition, be static. Even if "meta-rules" are specified to change the device's state-transition rules, the complete set of state-transition rules for the device is still pre-specified. The behavior of the device is still a necessary consequence of this enlarged set of pre-specified rules.

Formal-computational devices are designed to have reliable, replicable input-output behavior and as a result they are designed to operate despite environmental fluctuations. The representation of the structure-function dependencies of formal-computational devices (fig. 7.1) consequently has no causal links between the environment and the syntactic structure of the device. Thus formal-computational devices are *syntactically nonadaptive*.

A device could escape this restriction to fixed syntactic structure if new rules were implemented which were not necessary consequences of pre-specified rules. The new rules would have to be contingent upon events outside the device. However, such devices, whose syntaxes would be contingent upon interactions with their environments, could no longer be classified as formal-computational devices, since their behavior would not be completely stable. These *adaptive* devices will be discussed in chapter 9.

Specification. We have just noted that the behavior of completely programmable, computation universal formal-computational devices can be completely specified by their designer-programmers. In the course of operating these devices, all order is imposed by explicit rules and input symbols. No state-transitions or inputs can be left unspecified when the machine implements a computation. Everything must be laid out in advance, either through direct specification of the behavior (via lookup tables) or some sort of procedural specification (via algorithms). The state spaces and state transition rules can be completely known to the programmer.

What occurs in a computer model or simulation must be already encoded symbolically in a finite notation from the start; everything which takes place within the interpretive universe of the formal

system is inevitably bounded by the possibilities inherent in a particular representation language and the finite string lengths needed if the computational process is to be completed.

Closure. It is always possible to circumscribe the set of possibilities of a finite state machine within some finite notational system, by specifying all the possible mappings, in Ashby's notation of the preceding section, f: I x S into S.

We may not be able to physically write down a complete enumeration of the $2^{10,000}$ states of our 100 x 100 node cellular automaton (there are not enough molecules in the universe), but we can circumscribe all the possible states in a notation of a string of 10,000 characters in which each character may be either a 0 or a 1. The behavior of the cellular automaton is closed relative to this finite notational system. The complete set of possible input-output state transitions, f: A -> B, can be notated as the mapping of the set of these 10,000 character long input strings to an output set of these 10,000 character long strings. We can recognize all the possible state-transition behaviors of our cellular automaton in our finite notation of input-output mappings.

Formal-computational devices lack real world semantics

Our definition of computation, formal systems, and the behavior of formal-computational devices sees them completely as syntactic processes, entirely devoid of any semantics internal to the process or device itself. This is the formalist, "programs-are-pure-syntax" view :

> The *formalist* sense of computational psychology is more common than the mathematical sense. It covers those theories which hold that mental processes are, and/or are to be explained in terms of, the sorts of formal computations that are studied in traditional computer science and symbolic logic. These disciplines define 'computation' as *the formal manipulation of abstract symbols, by application of formal rules*. That is, the criteria for effecting one symbol-manipulation rather than another, and also for distinguishing the various transformations that are possible, are purely syntactic. (Boden, 1988, p.229)

> If computer science is anything it would seem to be the study of *formal* symbol manipulation, which I take it means that the symbolic ingredients are treated *without regard to their externally attributed semantical weight*. Just as a proof procedure is denied access to the interpretation of the sentences under its jurisdiction, and just as an adding machine has no access to the set-theoretic number that is encoded in the arrangements of its parts, so too the computer cannot behave in virtue of the reference of the ingredient constituents. Such a design would violate the foundational notion of computation. (Smith, 1986, p.45)

The semantics of formal-computational devices are not part of the device itself, since there are no causal relationships between the symbols and events in the world outside the device. If the symbols manipulated in a computer are to have any definite relation to the external world, it must be either through human interpreters or through sensors and effectors.

For computers without sensors or effectors, human beings decide what the symbols the devices manipulate stand for or signify. Human beings upon receiving the uninterpreted symbolic output of the device must decide how this symbolic output will be translated into actions on the real world. A given program can be interpreted any number of ways, as many ways as the human interpreter can think of. And the human being's interpretation is almost never a completely formalized relation to the external world. There are always tacit assumptions and unconscious connections involved in the interpretation. Human interpretation appears even more vague in character if we consider how a human being will act given the output of a particular program. This is not even approaching the issue of changing interpretations, or how we would decide, given the ambiguities involved, if an interpretation had changed.

Sensors and effectors are a different matter. Once we incorporate measurements and controls into our device we do give the device inherent, necessary, stable semantics, *but the device is no longer a formal computational device, since it could no longer be described solely in terms of computations.* We

would see measurement and control state-transition patterns in its observed behavior, as well as computations. We will discuss such formal-robotic devices in the next chapter.

The status of the semantics of formal-computational devices has been hotly debated within computational psychology. The clearest overview can be found in (Boden, 1988). This debate is really about whether programs have inherent, "internal semantics," and whether such a concept makes any sense at all. The debate is connected to disputes over the logical semantics of Carnap and model-theoretic accounts of semantics, which hold that semantics can be completely encoded. If semantics can be completely encoded, then there is no need for "external" semantics, some external connection with the world. Everything of relevance can be captured in a self-contained, closed, completely formalized set of words and word meanings.

The programs-are-pure-syntax view is held by Searle's (1980) "Chinese box example" and Fodor's (1980) "methodological solipsism," *contra* the programs-have-semantics view found in Newell's (1980) "physical symbol systems," Pylyshyn's (1984) "symbol level", and Haugeland's (1988) "interpreted automatic formal systems." A confusing tendency throughout the programs-have-semantics view is that it labels operations which appear to be completely syntactic with "semantic-sounding" names, e.g. Newell's operations of "designation" and "interpretation" (Newell, 1980, p. 156-158). If one closely examines the basis for an internal "semantics" of computers (e.g. in Newell, 1980 and Pylyshyn, 1984), it is apparent that it rests precariously on the dubious essential nature of the difference between a finite machine described as having a control box and tape (a finite tape Turing machine), and the same machine described without such a distinction. Thus whether a given token is a semantically-laden "symbol" depends upon which type of description we choose. Consequently, those who hold this view cannot criticize those of us who choose descriptions which do not support a separate "symbol level," thereby rejecting the programs-have-semantics view (see app. 2 for supplementary discussion).

Formal-computational devices are confined to operating within completely encoded problem domains.

Because formal-computational devices do not themselves make measurements, any problem which is to be dealt with via a computer program or simulation must be presented to it in a completely symbolic form. That is, the problem must be completely encoded or formalized before it can be solved by symbol manipulation. *This task must be accomplished by agencies outside the formal-computational device because it entails processes of measurement*. How easy this process is depends upon the nature of the problem.

Formalization of a problem domain is an art, since it consists of factors which are not themselves completely formalized, such as the choice of appropriate observables. Various kinds of intuitive insights, tacit images, informal analogies, and heuristic strategies must be used in these situations.

Once the nonformal aspects of the problem have been solved and the problem can be expressed completely in symbolic form, then formal-computational devices can be used to search for solutions. Thus, the tractability of a problem domain to computational solution depends upon whether these nonformal aspects of the problem can be coped with.

In problem contexts such as formal games and mathematical problems which are already completely encoded in notations which encompass the solutions, the primitive encoding is straightforward and all other transformations can be expressed in that language.

Unlike the domains of mathematics and classical physics, most complex, biological, psychological, and social systems have stubbornly resisted formalization. Nearly every problem involving human beings in everyday contexts has resisted reduction into rule-governed processes.

Examples of completely encoded domains are chess and checkers, in which the possible movements of the pieces on the board as well as the goal of the game can be readily expressed symbolically. Examples of an unencoded problem domain is recognizing and classifying various kinds of human

utterances, be they speech, music, grunts, or inadvertently produced sounds. Not that the latter domain cannot be effectively encoded in various ways, depending on the problem in mind, but that the nature of the domain itself is not symbolic in character, as it is with a game. It is no accident that artificial intelligence has had its greatest successes within symbolic or formal problem domains, and its least success in undefined, unencoded real world situations or nonformal problem domains (Dreyfus, 1979, p.292).

Of course, the tasks involved in developing an effective algorithm once a suitable encoding has been found are far from trivial. The development of chess programs rivalling all but the best human players, for example, have taken several decades of sustained effort on the part of highly talented and creative computer programmers.

The main use of formal-computational devices is to reliably execute complex, programmed symbol transformations for problems that have already been solved. The great bulk of computer programs in use today are used to keep and access records, to coordinate and process information. A smaller number are used to implement predictive models. A still smaller number are used for the purpose of simulating some problem domain and the effects of prospective solutions. These simulations show us the logical consequences of the assumptions that we put into them, and therefore, are invaluable for testing complex formal hypotheses.

Lady Lovelace and the limitations of computing devices

We have defined formal-computational devices and examined their major functional properties. In this light we can look at the debate over what computers can and cannot do. This debate can be seen to have started at the dawn of the computer age, even before a working computer had been built, by some observations made by Lady Lovelace in the 1840's.

Much of the subsequent debate has implicitly involved the issues of syntactic and semantic stasis and their lack of autonomous behavior. *The major limitations of formal-computational devices stem from their lack of inherent semantics and their nonadaptive structure*. We will concentrate on criticisms involving the essential *functional limitations* of formal computational devices, as opposed to "philosophical worries about whether machines could truly be said to *believe,* to *know,* to *mean*, to *understand*, to *feel*, and so on" (Pylyshyn, 1987, p.vii; see Flanagan, 1984, chap.6; Turing, 1950; Boden, 1988; Dreyfus, 1979; Searle, 1980; for general discussions).

Countess Ada Lovelace was a close friend of Charles Babbage, the inventor of the first mechanical automatic computing machine. In 1842 she wrote a series of Notes on Babbage's Analytical Engine, which was still under construction. In "Note G" she began,

> It is desirable to guard against the possibility of exaggerated ideas that might arise as to the power of the Analytical Engine." Ada recognized the dangers of discovery, first, the disposition to "*overrate* what we find to be already interesting or remarkable"; and then, in compensation, a willingness to "*undervalue* the true state of the case. (Baum, 1986, p.81; the quotation is from "Note G" of the Menabrea Scientific Memoirs, reprinted in Morrison & Morrison, 1961)

She continued,

> The Analytical Engine has no pretensions to *originate* anything. It can do *whatever we know how to order it* to perform. It can *follow* analysis; but, it has no power of *anticipating* any analytical relations of truths. Its province is to assist us in making *available* what we are already acquainted with. (Lady Lovelace, quoted from Baum, 1986, p. 82)

Lady Lovelace's remarkably prescient observations bear on the problem of specification, the status of computationally generated truths, and the problem of generating new primitives through computations. We will encounter each of these fundamental limitations as we examine each of these problems.

The specification problem

Because of the static nature of programmable, computational devices, everything which happens within the system must be specified in advance. This is obvious to anyone who has ever written a computer program or simulation. The computer "can do *whatever we know how to order it* to perform," but we must know how to order it to perform the desired behavior we want. This is the *problem of specification.*

The problem of specification is the problem of defining and encoding all the relevant contingencies and what actions are to be taken in each contingency, of specifying exactly what the computer must do. Even for the simplest real world systems, this can be a monumental task, when it can be accomplished at all. If many aspects of the problem or task involve tacit assumptions, the specification problem may prove intractable for that particular problem domain (Dreyfus, 1979, 1980, 1986).

The logic-based artificial intelligence research program holds out the hope that this specification problem has an end, that eventually all aspects of human situations can be dealt with if only we encode enough of the domain, and that the problem will then disappear. In addition, strong AI assumes that biological organisms and human cognitive mechanisms operate much the same way that computers do, so that there *must* be computer programs isomorphic in structure to their natural counterparts (Searle, 1980). Hubert Dreyfus has identified and criticized a number of these assumptions:

> All artificial intelligence work is done on digital computers because they are the only general-purpose information-processing devices which we know how to design or even conceive of at present. All information with which these computers operate must be represented in terms of discrete elements....The machine must operate on finite strings of these determinate elements as a series of objects related to each other only by rules. Thus, the assumption that man functions like a general-purpose symbol-manipulating device amounts to
> 1. A biological assumption that on some level of operation -- usually supposed to be that of the neurons -- the brain processes information according to formal rules.
> 2. A psychological assumption that the mind can be viewed as a device operating on bits of information according to formal rules....
> 3. An epistemological assumption that all knowledge can be formalized, that is, that whatever can be understood can be expressed in terms of logical relations, more exactly, in terms of Boolean functions, the logical calculus which governs the way bits are related according to rules.
> 4. Finally, since all information fed into digital computers must be in bits, the computer model of the mind presupposes that all relevant information about the world, everything essential to the production of intelligent behavior, must in principle be analyzable as a set of situation-free determinate elements. This is the ontological assumption that what there is, is a set of facts each logically independent of all the others. (Dreyfus, 1979, pp.135-136)

What artificial intelligence has never dealt with are the processes by which "all knowledge can be formalized." In making good on the claims of the ontological and epistemological assumptions (3 & 4 above), the specification problem is directly encountered. If the claim is made that a given phenomenon is reducible to rules, it is incumbent upon those making the claim to show specifically what rules are involved, else the claim is gratuitous. This would necessarily involve the semantic processes of encoding and decoding, of measurement and control, of perception and action. Precisely at the same time physics is becoming self-aware of the irreducibility of these epistemological primitives, artificial intelligence is ignoring all but the formalism. In doing so artificial intelligence has committed the same errors of classical physics in advocating the informational equivalent of a crude Laplacian determinism.

What solutions does Dreyfus propose? His somewhat vague remedies involve emphasizing the role of the body in mediating between the mind and the world as well as the role of nonsymbolic, analog Gestalt (Dreyfus, 1979, pp. 240-241) or holographic processes (Dreyfus & Dreyfus, 1986). In a sense the introduction of the body is related to the program advocated here, if the body can be seen as the totality of semantic operations of measurement and control as well as noncognitive, nonsymbolic interactions with the environment. But even addition of fixed semantic operations, the construction of robots, is not enough, as we shall see in the next chapter.

A third and very promising strategy is to introduce adaptivity by designing devices with their human users in mind. The human-machine interface is developed so as to *support* human abilities (Dreyfus, 1986, p.191). Human-machine cooperation would take the best from both worlds: the ability of human beings to adaptively define their own semantic relations and the ability of computers to manipulate complex strings of symbols. This is a very positive direction for the design of computers, but it does not make the machines themselves more autonomous in solving the specification problem. These devices, relying on the adaptability of their human part, are no longer formal-computational, but adaptive and/or evolutionary in their behavior.

The frame problem

For the most part the computationalists have not responded directly to the bulk of Dreyfus' criticisms. One exception is the frame problem (e.g. Pylyshyn, 1987). Related to the general problem of specification is the problem of deciding what aspects of a situation are relevant to a particular problem. What is the appropriate encoding? What kinds of representations are needed? The existence of an effective procedure to solve this problem rests on the assumption that the entire context can be effectively encoded into a finite number of discrete properties and that all the relevant interrelationships between the properties can be known (Dreyfus' epistemological and ontological assumptions, cited above). The computationalist responses range from the deductivist strategy of encoding more of the world from the beginning and using logical deduction, to the nondeductivist, selectionist strategy of generating formal alternatives and selecting those which seem to work (Janlert, 1987). Throughout this spectrum, however, the artificial intelligence programs make the assumption of a model-theoretic "closed-world" in which a fixed set of properties is dealt with. Even the nondeductivist, selectionist strategy eventually runs out of formal alternatives, and a human programmer must intervene to enlarge the set. The inability of formal-computational devices to create their own new primitives lies at the root of this problem.

The problem of the generation of new primitives

Lady Lovelace's observation is that the Analytical Engine "has no pretensions to *originate* anything" bears on the question of whether computers can be creative, whether they can generate new syntactic relations that were not in some sense already programmed into them. The reason why computers themselves cannot be programmed to solve the specification problem is that they cannot generate new symbol primitives, either syntactically or semantically. Were they able to generate their own primitives, the specification problem would go away, as human beings would no longer have to completely specify the terms of a given problem's definition and solution.

A syntactic primitive would appear as a new primitive symbol, not reducible to the actions of other symbols. The creation of a new syntactic primitive would also entail new syntactic rules to take the new primitive into account, not reducible to existing rules.

A semantic primitive would be a new means of interpreting a given symbol. Because formal-computational devices in and of themselves have no external semantics, the whole question of new semantic primitives is moot until we discuss devices which do have inherent external semantics.

Machines as generators of new ideas

The static nature of formal-computational devices precludes the possibility of generating new syntactic or semantic primitives. In terms of computational creativity, the best one can do with formal-computational devices is to build up complex categories out of the intersections and unions of the pre-existing ones, searching for particularly useful combinations.

Such a strategy is exemplified in the "mathematics discovering" program, Automatic Mathematician (AM), of Douglas Lenat (1978, 1982). Lenat's theory of intelligence holds that all problem-solving can be cast in terms of searches within fixed search spaces.

> This program starts with a collection of primitive mathematical (set-theoretical) concepts, and some transformational rules (*heuristics*) which it uses to explore the space potentially defined by the primitive concepts. It generates and then explores concepts and hypotheses about number theory, guided by its hunches about which are likely to be the most interesting. (Boden, 1988, p. 209)

Lenat specified on the order of "100 elementary mathematical concepts (e.g. sets, composition of operations) and about 230 'heuristic rules.'"(Richie & Hanna, 1984, p. 251) The heuristic rules steer the machine through the search space and also determine the "interestingness" of a particular pattern. He let the machine run through combinations of operations on objects, noting any interesting patterns. The usefulness of the patterns found are questionable in many cases, "some of AM's heuristics are 'suspiciously specific'" (Boden, 1988, p.210), many of them apparently *ad hoc* in design (Ritchie & Hanna, 1984, section 4.3). Nevertheless, the program clearly illustrates the only strategy available for formal-computational devices to generate "new" ideas.

Lenat claims

> The AM project has demonstrated that open-ended scientific theory formation (the defining and exploring of new concepts and relationships) could be mechanized, could be modelled as heuristic rule guided search, using a few hundred heuristics for guidance. (Lenat, 1978, p.278)

To my knowledge, Lenat's program has not been criticized from the standpoint of its inability to create new conceptual primitives. The AM program clearly cannot create concepts outside of combinations of primitives that it already had. The "new concepts" are not, in fact, "new," but set-theoretic combinations of old, pre-existing ones. Lady Lovelace's observation, cited above, seems apt here: "It can *follow* analysis; but, it has no power of *anticipating* any analytical relations of truths. Its province is to assist us in making *available* what we are already acquainted with." Lenat's device is not open-ended, since it would be very simple to devise a finite, closed notation describing all the possible combinations of primitive concepts. More will be said about computationalist conceptions of novelty in chapter 12, where issues of emergence and open-endedness will be discussed at length.

Purely digital devices cannot self-complexify

Underlying the inability of formal-computational devices to create new primitives is the digital nature of their construction, which is necessary for the degree of reliability they must exhibit. An important difference between formal-computational devices and biological organisms is that biological organisms rely on mixed digital-analog processes while formal-computational devices are constructed from purely digital organs (see app. 5).

> For the present, we wish to recognize that where continuous dynamic processes are artificially suppressed, as in formal automata theory or in a computer model of some aspect of cognition, the intrinsic generation of new primitives is precluded. A system executing *solely* in the discrete mode cannot self-complex. The general argument is that any system whose present competence is defined by a logic of a certain representational power cannot progress *through operations in the discrete mode* to a higher degree of competence (e.g. Fodor, 1975).
>
> Suppose that the operations in the discrete mode are the projection and evaluation of hypotheses. A hypothesis is a logical formula, as is evidence for its evaluation, and both formulas must be expressed in the discrete symbols of the system's internal language. If the evidence is sufficient to confirm the projected hypothesis, then the fact to which the hypothesis corresponds can be registered in the representational medium. Importantly, however, the range of hypotheses projected and the range of evidence considered are both restricted to the expressive range of the symbols available to the system. Any hypothesis or any evidential source that must be expressed in symbols other than those available cannot be entertained. In sum, a system executing solely in the discrete mode cannot increase its expressive

power. It cannot develop the capacity to represent more states of affairs at some later date than it can represent in the present. What it can do is to distinguish, within limits, states of affairs that occur from those that do not. The order of complexity achievable by a system executing solely in the discrete mode is frozen; it is determined by the order of complexity with which it began. How is the order of complexity raised in a system with no continuous dynamic processes such as a computer? By coupling it to an external intelligent device (a programmer) that writes in new symbols and discrete rules.

To summarize, when information used by a system is construed linguistically (that is, ignoring the relationship between symbols and dynamics), it cannot spontaneously increase in expressive power. In order to do so, such a system would have to be endowed with preadaptive foresight, possessing predicates that are currently useless but that will be relevant someday. Since such foresight is not possible, computational models are limited to the order of complexity with which they began. They cannot outperform the control rules which govern their operation (Tomovic, 1978). Natural systems, on the other hand, are open to complexity and require a construal of control information that is self-complexing. (Carello *et al*., 1984, p.233)

Natural systems possess self-complexing properties by virtue of their analog processes, which cannot be fully captured by static formal models.

Conrad (1983) presents a slightly different argument based upon the gradualistic premise that nearly continuous changes must be possible for evolution to occur.

Systems cannot bootstrap if they are constructed from standard components, where the number of possible types of components is limited. Technical information processing systems such as present-day digital computers are of this character. They are constructed from a limited number of types of standard switching elements. Unlike the proteins of biological systems -- the real switching components in biology --these standard components cannot be gradually transformed. In fact they cannot be transformed at all once the system is constructed. They are the *given* primitives. The only way a system constructed from such primitives might conceivably be more amenable to evolution is by organizing the interconnections among them differently. However, systems of standard building blocks have a powerful feature which is incompatible with amenability to evolution. The powerful feature is *structural programmability* : it is possible to effectively design them so that they realize any rule which can be expressed in terms of a computer program. (Conrad, 1983, p.233)

In terms of the concepts discussed earlier in this chapter, Conrad's concept of a structural-programmable device corresponds to one which is both completely programmable and computation universal.

Drexler (1989) also draws a very similar distinction to Conrad's, between devices with adaptable parts and devices with stable ones, which he calls O[rganism]-style and M[achine]-style organizations respectively. Like Conrad, Drexler argues for the advantages of a gradualist, O-style evolvability vs. the inflexibility of M-style devices and their susceptibility to catastrophic error. "... M-style systems suffer constraints that effectively eliminate significant evolutionary moves" (p. 511). Nevertheless, he advocates M-style nanoreplicators over O-style ones because they would be simpler to construct and because evolvability is not one of his more important design priorities.

While the availability of small structural modifications leading to small functional modifications is probably important for the rate of evolution, the relative absence of gradualist pathways would not preclude evolution entirely. The important point is that purely computational devices do not construct or modify their primitives, and this does foreclose the possibility for fundamental novelty.

The language acquisition debate

Argument involving the formation of new primitives also arose in the debate between the interactionist view of Piaget on one hand and the innatist view of Chomsky and Fodor on the other regarding the learning of language (in Piatelli-Palmarini, 1980). The rationalist, innatist view sees language acquisition as an epigenetic, rule-governed generation of the rules for language understanding and use. Piaget, on the other hand, sees language as an interactive, adaptive construction involving significant interactions between the organism and the environment.

One can ask, even if language capability is ontogenetically innate, how did it evolve in the first place? What mechanisms were involved, and why couldn't there be analogous mechanisms operating in the brain? The typical response on the part of the rationalists is to "pass the buck to the biologists" by declaring that such considerations stand outside their theory.

Analogously, the rationalist position begs the question of how more powerful logics are created from less powerful ones over the history of mathematics, if all ideas are innate and logical in character (Piaget, 1980). Fodor (1980) makes the argument that the learning of new conceptual primitives would correspond mathematically to the computation of more powerful logics from less powerful ones. Since this is demonstrably not possible, then either human beings are not solely computational in nature (Piaget's conclusion) or if they are, then concept learning in that sense is not possible (Fodor's conclusion).

At some point either in the historical development of mathematical concepts or in the evolution of language capacity, noncomputational processes must be invoked.

> [The innatists] contend that the study of the nature of the unfolding of the genetic program is a study *on a very different level* from the investigation of how this genetic endowment reached its present state in the species. They cannot see any contradiction inherent in this methodological stand. The tenants of the order-from-noise principle purport, however, to demonstrate that no explanation in terms of preset programs is *logically* sound unless one gives at least a hint of how programs are assembled at their source. Sweeping the dust under the rug (that is, pretending not to bother with the origins of programs) does not improve, in the long run, the tidiness of the innatist clubroom. The problem will have to be met head on one day or another. Self-organization, so their story goes, has to precede, logically and factually, programmed regulation. (Piatelli-Palmarini, 1980, p.19)

New primitives, if they are to appear must be created through noncomputational processes, processes which stand outside the scope of purely computational devices.

Lucas' Gödelian argument

Related to the inability of formal systems to step outside of themselves, to generate new primitives is the argument of Lucas(1961). Granting the existence of potentially infinite Turing machines, Lucas argues that the static nature of formal systems is qualitatively different from the dynamic nature of human reasoning:

> Gödel's theorem must apply to cybernetical machines, because it is of the essence of being a machine, that it should be a concrete instantiation of a formal system. It follows that given any machine which is consistent and capable of doing simple arithmetic, there is a formula which it is incapable of producing as being true -- i.e., the formula is unprovable-within-the-system -- but which we can see to be true. It follows that no machine can be a complete or adequate model of the mind, that minds are essentially different from machines. (Lucas, 1961, p.44)

Lucas demonstrates a procedure by which the initial assumptions and operations of a given machine could be written down by a human being as axioms and rules of inference for an equivalent formal system.

> The conclusions it is possible for the machine to produce as being true will therefore correspond to the theorems that can be proved in the corresponding formal system. We now construct a Gödelian formula in this formal system. This formula cannot be *proved-within-the-system*. Therefore, the machine cannot produce the corresponding formula as being true. But *we* can see that the Gödelian formula is true... (Lucas, 1961, p.47)

The typical computationalist counterargument is to assert that a more complex machine can be constructed which will include the enlarged set of theorems recognized by the human being (e.g. Lucas, 1961, p.48-49; Hofstadter, 1979, p.471-475), *but this is allowing the formal-computational device a dynamism it does not have when unaugmented by human beings.*

It is agreed that a particular Turing machine can simulate any piece of human behavior once it has been encoded, but this does not mean that a particular machine can simulate *all* of human behavior.

> [Turing] argues that the limitation to the powers of a machine do not amount to anything much. Although each individual machine is capable of getting the right answer to some questions, after all each individual human being is fallible also: and in any case 'our superiority can only be felt on such an occasion in relation to the one machine over which we have scored our petty triumph. There would be no question of triumphing over *all* machines.' But this is not the point. We are not discussing whether machines or minds are superior, but whether they are the same. (Lucas, 1961, p.49)

In other words, human minds are plastic and adaptive, while computational machines are not. Turing's argument regarding "triumphing over all machines" (Turing, 1950, p.16) rests on fallaciously attributing properties of concrete, material individuals making up a class of things (i.e. particular Turing machines) to the abstract class itself (the class of all Turing machines). This is exactly the "platonic fallacy" criticized in Goodman (1956). Turing's argument is equivalent to saying that "there is no contradiction between the fact that for any natural number there can be produced a greater number and the fact that a number cannot be produced greater than any number." (Lucas, 1961, pp. 47-48). In the case of all possible Turing machines and natural numbers, the complete class itself can never be physically realized.

Ultimately, this is another disagreement between synchronic and diachronic views of the world, between static, all-encompassing time-independent descriptions and changing, unfolding, time-dependent phenomena. This is *not* an issue of whether "time" is represented as a variable in a model, but whether the model itself changes over time. No formal model, by definition, changes its structure over time.

It is clear that Turing (1950) assumed that human beings, like formal models, are fixed in their capacities, while Lucas (1961) assumed a human being who is capable of learning and creating new categories. The synchronic theorist, when confronted by change, merely adopts another enlarged synchronic theory until that one is superceded, and so on. The diachronic theorist is forced to build into the theory the possibility of change. But even diachronic formal models themselves are fixed, synchronic entities while the physical world is changing. The formal model itself only implements rate-independent relationships. To get truly diachronic (emergent) behavior, one must construct physical devices which possess rate-dependent properties unspecified by any model.

Chapter 8: Formal-robotic devices

> Mechanical imitations of certain human functions have been with us for centuries. Many medieval clock towers are equipped with mechanisms that mark the hours with elaborate morality plays enacted by mechanical saints, knights, bishops angels, demons and all kinds of animals. Smaller devices that walked talked, swam, breathed, ate, wrote with quill pens, or played musical instruments have amuse polite society since at least the fifteenth century. ... These early clockwork machines--powered by running water, falling weights, or springs--copied the motions of living things, but they could not respond to the world around them. They could only *act*, however charmingly.
>
> Electrical, electronic, and radio technology, developed early in this century, made possible machines that could *react* --to light, sound, and invisible remote control. The result was a number of entertaining demonstration robots--as well as thoughts and stories about future humanlike machines. But only simple connections between the sensors and motors were possible at first. These newer machines could sense as well as act, but they could not *think*. (Moravec, 1989, p. 6)

In the last chapter it was argued that purely syntactic, computational devices, because of their fixed structures and environment-independent operation, are completely nonadaptive in nature. In this chapter computational devices are connected to their environments via stable sensors and effectors, thus conferring upon them fixed external semantics. We thus have what are commonly known as robots. Do robots as a consequence of the addition of sensors and effectors have capacities and limitations which are different from computers? Does the addition of external semantics change the nonadaptive nature of the device?

It will be argued that even though the addition of external semantics is exceedingly important, allowing formal-robotic devices to perform outside of purely symbolic domains and in the material world, the operation of sensors and effectors does not change the structure of the device over time and experience. Thus the structure of the formal-robotic device is still completely independent of its environment, and the device is still completely nonadaptive.

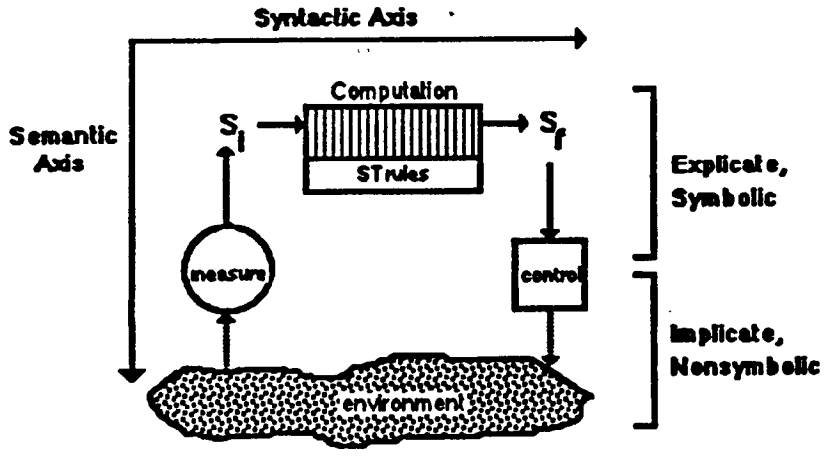

Figure 8.1 Syntax and semantics in a formal-robotic device.

Definition of formal-robotic devices

Formal-robotic devices are formal-computational devices which are connected to their environments via operations of measurement and control. The structure-function relations for such devices are shown in figure 8.1. A formal-robotic device's operation involves a sequence of measurements, computations, and controls. Unlike formal-computational devices, the input states of formal-robotic devices are contingent upon events happening in their environments, and the outputs of such devices have direct effects on the world outside the device.

Formal-robotic devices are distinct from formal-computational devices in that they are not purely *computational* in behavior. Formal robotic devices necessarily involve noncomputational processes of measurement and control. There is a tendency in computer science and artificial intelligence to ignore or trivialize the distinction between computers and robots. Often this is due to the lack of a principled distinction in those fields between computations on one hand and transducers (both input and output types) on the other. As often it is due to the feeling that computation is the central, most important function and connections of the world are literally "peripheral" to computation.

A observed state-transition diagram for a simple formal-robotic device is shown in figure 8.2. The device includes its sensors and effectors, but the observed state-transitions for measurements and controls straddle the device-environment interface. On the device side are the symbolic states, the results of measurements and the inputs to control processes, while on the environment side there are nonsymbolic states, the precursors of measurements and the effects of control processes.

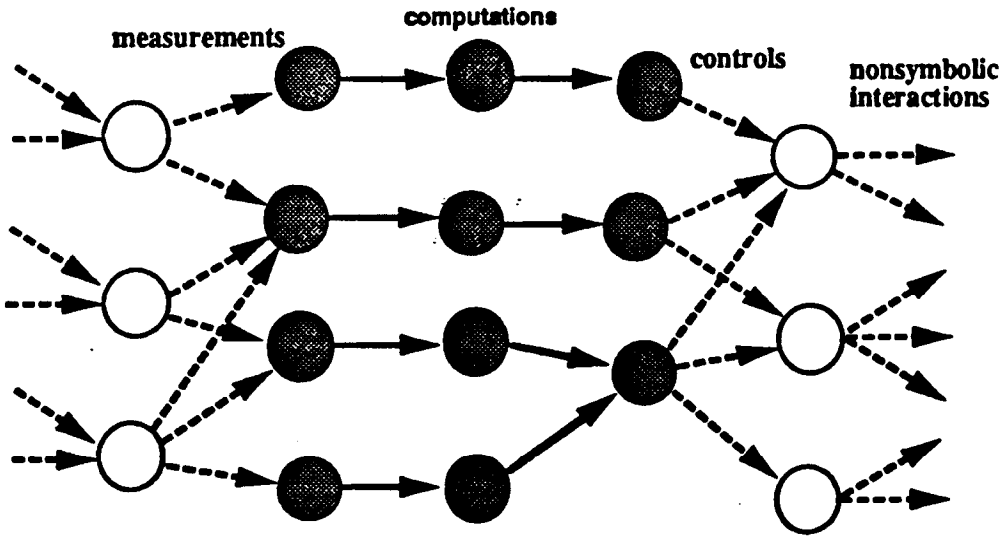

Figure 8.2 Observed state transition diagram for a simple formal-robotic device.

The observed state-transition structure of these devices is qualitatively different from formal-robotic ones, in that *the computational states of the device are causally connected to the external world via noncomputational, semantic linkages*. Measurement and control transitions are necessary to describe the interactions of this device with its environment.

One could make the objection that the device's environment could have purely computational processes, as an artificial animal operating the artificial environment of an evolutionary simulation. To do this would functionally reduce the device to syntactic computations, effectively eliminating any measurements or controls. In the classical mode of analysis of chapter 4, if all states are part of computations, the simulation will appear to us to completely describable via rate-independent linkages--i.e. *it will appear to us as one large computation*. No rate-dependent terms would be necessary for our dynamical description of that system. We would not see the relation of the device to its environment in terms of measurements or controls. The interface between a formal-robotic device and its real world environment and an artificial animal and its simulated environment are thus clearly different in the classical view, regardless of our observational frame. In the former the interface is describable by mixed rate-dependent/rate-independent linkages, i.e. measurements and controls, where in the latter the interface is describable only utilizing rule-governed rate-independent, symbolic constraints, i.e. computations.

From the observer-centered perspective, the only way to make the artificial life simulation look to us like a formal-robotic device would be to redefine the observational frame so that the device-environment interface was no longer composed solely of computational state-transitions. To do this we would have to discard much of our knowledge of the artificial environment by combining observed states, such that the artificial environment and the organism-environment interface would cease to appear to be computational. *It would be the equivalent of ignoring information that we already had.* Where previously we had a deterministic model, we have eliminated observables such that the model now is dependent upon nondeterministic state transitions. It would be like making some of the states of a (previously state-determined) computer unobservable and then declaring that the device is a nondeterministic one. It is not clear why we would ever want to do this, why this would be useful, except as an exercise. While the artificial animal would then appear to us as a formal-robotic device, the *simulation as a whole* would cease to appear as a formal-computational device. Either the artificial animal with its artificial environment are seen as a large computational simulation or the animal is seen as a formal-robotic device and the environment is seen as a network of nonsymbolic interactions. Both interpretations are not simultaneously possible. *Within a given observational frame, the observed behaviors of formal-computational devices and formal-robotic devices are mutually exclusive.*

Examples of formal-robotic devices

Biologically, formal-robotic devices correspond to phylogenetically primitive individual organisms incapable of learning of any sort. Here the functioning of sense organs, the coordinative apparatus, and the effector organs are all determined genetically, and these are not altered in any way by the experience of the organism. One must go very far down the phylogenetic tree to find this level of nonadaptivity in the biological world: even flatworms are capable of very rudimentary learning.

Formal-robotic artificial devices would include pre-programmed robots, completely automated assembly lines, pre-programmed classifier devices, pre-programmed numerically-driven machine tools, and formal scientific models with their associated measuring devices, as well as a host of deterministic sensing-deciding-acting mechanisms.

When a human interpreter is included in the description of the device, the human interpreter assumes the role of both sensors and effectors. In essence, the human being performs measurements which become inputs for some computational process and takes actions based upon the outputs of such a process. Traffic safety analysts, for example, take observations made by human beings regarding the numbers and types of accidents, enter them into their simulations, examine the results, and take action to improve traffic rules, change enforcement levels, or alter the construction of roads and vehicles.

For example, if we look at a formal bureaucracy, human beings generally interact with the outside world and encode their interactions in a particular way, apply formal rules or "policies," read out the result, and act accordingly. Many loans approval procedures are fully formalized in this way: there is an encoding of the applicant's characteristics, a computation is performed, and the action of the financial institution is taken as a necessary consequence.

Including humans in the definition of the formal-robotic device is only valid if the human beings implement stable measurements and controls and if they do not alter the computations being performed. Either condition would violate the stability of the device's structure, and we would no longer have a formal device,

Fixed syntaxes and semantics

The sensors and effectors associated with these devices are fixed and stable relative to their environments, permitting reliable measurements and actions. Formal-robotic devices thus have fixed syntaxes and fixed, nonarbitrary semantics.

Syntactically static. For the same reasons formal-computational devices are syntactically static, so are formal-robotic devices. The computational part of both devices implements a formal system, operating reliably and independently of events in the environment. Neither fixed computer programs nor the types of fixed robots discussed in this chapter changes its internal computational structure in response to experience.

Semantically static. Formal-robotic devices, despite the fact that they do themselves have external semantics, still have fixed semantics. Their interaction with the world does not change the semantic relations their sensors and effectors implement. How can we verify this criterion? How do we know our sensors and effectors are implementing the same semantic relations between the symbols in the computational part and the world at large? How do we ascertain the stability of our model-world linkages?

There are basically three ways. We will consider the problem in terms of the stability of the measurement relation. The control relation would generally have similar modes of calibration.

The first method is to examine the probabilities of the observed state-transitions. Obviously if a completely new state transition is observed between a nonsymbolic and a symbolic state, a different measurement is occurring. Likewise, if the relative probabilities of the alternative state transitions involved in the measurement change, it will be increasingly less likely, from a statistical point of view, that the generative probabilities are the same.

The second method is to use a consensus of reference measurements. A set of measuring devices is constructed and calibrated so that they always agree on a particular measurement. The outputs of the sensors of the formal-robotic device can always be compared to this reference population, so that stability of the measurement operation relative to the reference population can be assured. Here for the comparison of controls, measurements would have to be made on the effects of the controls and the results of the measurements would be compared.

The third method is pragmatic, relying on whether the putative change in the measurement operation has changed the performance of the device in any way. If not, then relative to the device's performance, the measurement operation is equivalent relative to the performance level.

All three criteria form discrete equivalence classes, grouping measurements together which have the same symbolic results, but will be different on some finer level of resolution. None of the criteria fully capture the measurement process in the sense of producing a rule which can replace the measurement. One of the dangers is that when we form these equivalence classes, we often lose sight of the underlying, dormant differences. These criteria will be useful in chapter 10 when we discuss devices that change the measurements and controls they make and in chapter 12 when we discuss how we would recognize that a measurement and/or control had changed.

Capacities and limitations of formal-robotic devices

By virtue of their sensors and effectors, formal-robotic devices can carry out measurement and control operations in addition to computations. Not only can symbols be manipulated according to rules, but the machine can perform classifications on the world via its sensors and act on the world via its effectors. Formal-robotic devices unlike their computational counterparts are not limited to operating in completely encoded domains. The range of activities that sensors and effectors permit drastically enlarges the scope of what these devices can do as a class. *It is difficult to imagine any fixed function which could not be carried out by appropriate combinations of measurements, computations, and controls.*

Unlike formal-computational devices, the symbols of formal-robotic devices do have fixed, inherent real world semantics. These semantics are determined by both the sensors and the effectors of the device. The semantic categories within which the device operates are thus fixed as long as the sensors

and effectors are implementing the same semantic linkages, that is as long as they remain physically stable.

Performance characteristics

Because of the static nature of formal-computational devices, their behavior does not change in response to experience. Consequently the performance level of formal-computational devices is static (Gause & Rogers, 1981). The system is either "intelligent" if it performs well or "inane" if it performs less well. Such systems will not improve their performance without human intervention to change their state transition functions. There are no information channels feeding back into the system itself to alter the structure and subsequent behavior of the system. Such systems are completely non-adaptive.

Strictly speaking, because formal-computational devices do not have real world semantics, they do not produce real world performances. Performance is independent of time (experience) because the syntaxes and semantics of these devices are fixed; they do not learn from their past performance. The performance can either be good or bad, "intelligent" or "inane" but adaptation will not take place unless a human being intervenes to modify the program. The device itself cannot modify its own structure as a result of experience.

The necessity for adaptivity

Like formal-computational devices, formal-robotic devices must be specified by their designers. This specification includes not only the behavior of the computational part, but the means for building and calibrating the sensors and effectors.

W. Ross Ashby's essay, "Can a Mechanical Chess-Player Outplay its Designer?" poses the question, "to what extent is a machine restricted by the limitations of its designer?" To do this, he introduces a postulate to be disproved, in the form of Descartes' dictum: " ...there must be as much reality and perfection in the cause as in the effect." He says,

> Our working hypothesis is that we can eventually build a "real" brain. But for it to be a "real" brain it must produce cleverness of its own and not just give us back the ingenuity we have put into it. Here the truth or falsity of Descartes' dictum is crucial. If it is true, we are wasting our time; only if it is false can we succeed. (Ashby, 1952b, p.281)

He shows using an information-theoretic analysis that this dictum is true for devices which are not capable of learning from their environments. He demonstrates that Descartes dictum is not true, however, for those devices which generate alternatives and then perform some selection process based upon performance.

> We can see, for instance, that the more minutely we design a machine to play chess just as we want it to-admitting no other information-the more certain it is to play just *our* sort of chess, with all of our faults and wrongly conceived strategies. We are, in fact, in exactly the same position as the father, a keen but mediocre chess-player, who wants his son to become world champion. It is true that the father should teach the child, but he must not teach his son every reply in detail lest he limit the son's play to being the merest replica of the father's. If the father really wants the son to beat him he must sooner or later stop telling the son what to do and must send him out into the world to be subjected to all sorts of unselected experiences. The understanding father will not try to teach his son all chess but will try to teach him how to profit by future experiences.
>
> Designing a machine has much in common with teaching a child, for in each case the almost infinite possibilities have to be reduced to a selection. Were Descartes' dictum to be stated in the form "no child can know more than he has been taught" we would at once see the equivocation, for the teacher can not only teach the child facts but can also teach him how to use the "free" information in the world, and thus how to surpass his teacher. If we wish to build a machine that can beat us at chess, or to build a "real" brain, we must follow the same method; we must aim in design, not a machine that will play chess, but at a machine that can make trials, and select. The homeostat was intended to be a first step in this direction. (Ashby, 1952b, p.291)

Ashby's general strategy for building an "intelligence-amplifier" is to provide the device with a very large repertoire of possible stimuli and responses and let the machine select those which best accomplish a particular task. This strategy will be applied in the next two chapters to the generation and selection of alternative computations and alternative construction and selection of measurements and controls.

Chapter 9 Adaptive devices

> We are proposing here a model of a process which we claim can adaptively improve itself to handle certain pattern recognition problems which cannot be adequately specified in advance. (Selfridge, 1958)

> Many of the models 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 to answer the first. (Rosenblatt, 1962, quoted in Dreyfus & Dreyfus, 1988).

With adaptive devices, we come to the first devices whose internal structure is dependent upon its interactions with the external world. Adaptive devices modify their computational part in order to improve their performance in their environments. In adaptive devices pragmatic relations select for appropriate syntactic relations. Thus, with adaptive devices only a range of possible input-output functions need be specified for the computational part, thereby allowing the device to cope with unforeseen circumstances, thereby freeing the designer from having to foresee all possible contingencies. However, adaptive devices, too, have their limitations. These lie in the static nature of their semantic relations: all that an adaptive device can do must be accomplished within the fixed set of observable features and performable actions available to it.

Adaptive devices

An "adaptive device" is a formal-robotic device whose syntax is adaptive. The syntax becomes adaptive by making the computational part of the device modifiable through experience. A performance measure alters the computation of the coordinative part (see figure 9.1). A feedback loop between the environment and the internal structure of the

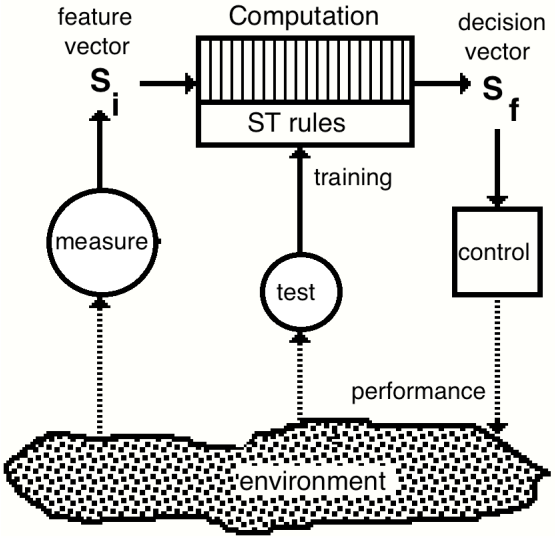

Figure 9.1 Structure-function schematic for an adaptive device. The behavior of the computational part of the device is dependent upon its real world performance, as assessed through an evaluative test procedure. Although adaptive devices have performance-contingent syntactic operations, their external semantics are fixed by their fixed sensors and effectors.

coordinative part is thus established, so that the computation performed by the coordinative part can be made contingent upon the environment, making learning within that part possible. Consequently, adaptive devices can learn from their environments and improve their performance levels within those environments in a way which sets them apart from "intelligent" but completely fixed programs (Gause & Rogers, 1982).

The computational part of an adaptive device can have all the complexity of formal-computational and formal-robotic devices. Whatever can be designed as a static program can be redesigned as an adaptive one by making the appropriate parameters contingent upon performance. Very complex input-

output functions incorporating higher level schemata, model-theoretic representations, logical operations, as well as "anticipatory" (Rosen, 1985) or "projective" capabilities (Campbell, 1985) are possible. Given rich feature and action spaces, adaptive organisms and devices are capable of a very high degree of behavioral flexibility.

What can be learned need not be specified. A programmer can reduce the specification problem by handing various decisions over to the adaptive faculties of the device and allowing the device to discover those computational mappings which are the most useful. For adaptive devices, the programmer needs to specify a range of alternatives rather than specific ones for specific contingencies. This is particularly useful when one is confronted with situations which are not known in advance, where it would be difficult or impossible to completely design the appropriate computation.

Once the adaptive device has found an appropriate computation which adequately solves the problem at hand, then the machine can be frozen by disconnecting the feedback from the device's performance to its internal structure, i.e. by disabling the training rules. The adaptive device is thus transformed into a formal-robotic one by removing degrees of freedom from its behavior.

Syntax, semantics, and pragmatics of adaptive devices

Adaptive devices consist of three sets of components, corresponding to semantic, syntactic, and pragmatic relations.

First, semantic relations are implemented via a set of sensors which perform measurements constituting the primitive "features" the device will utilize and/or a set of effectors which perform control interactions with the environment constituting the "actions" in its repertoire.

Second. the syntactic relations are implemented via a computational part which takes the value of the features as its input. The computational part is divided into two subparts, the state-transition algorithm, whose output is contingent only upon the features presented it, and the training-rule part, whose output is contingent only upon the performance measure.

Third, the pragmatic relations are implemented via a performance measure, which is a sensor separate from those which detect features. In an adaptive device, the pragmatic relations perform selective operations on the syntactic operations that are carried out.

Adaptive devices and formal-robotic devices

The difference between an adaptive device and its formal-robotic counterpart having the same features and/or actions lies within the performance-contingent part of the computation.

Broadly speaking, a formal-robotic device always implements the same input-output function in its symbolic, computational part. The behavior of this part is completely pre-programmed and replicable, a necessary consequence of the fixed state-transition rules of the computational part. No knowledge of the performance of the device is necessary to replicate this input-output behavior.

An adaptive device, on the other hand, has an input-output function which is not completely pre-programmed. The behavior of an adaptive device is not a logical consequence of its rules and initial conditions; its syntactic structure will be contingent upon its past performance. Consequently, knowledge of its performance is necessary to replicate the behavior of its computational part, but this performance measurement is itself contingent upon interactions between the device and its environment.

Adaptive classifiers and controllers

To be labelled "adaptive," a device might not necessarily have both sensors and effectors. Adaptive classifiers and adaptive controllers are two such possibilities.

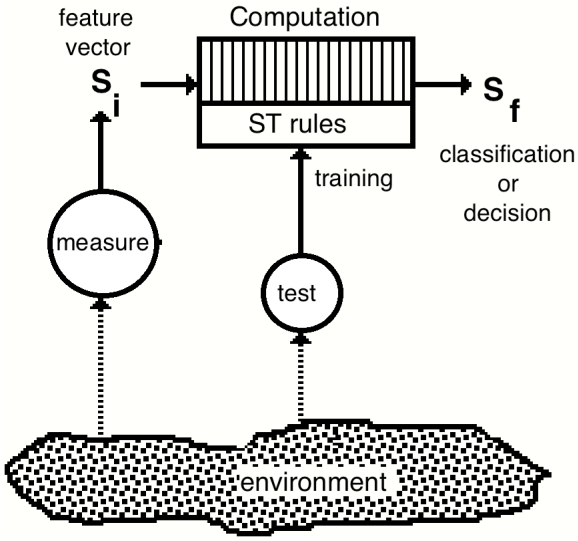

Figure 9.2 Schematic for an adaptive classifier. The device receives an input feature vector from the world, partitions the vector space through a computation, and is modified according to its performance.

Adaptive classifiers. An adaptive classifier (figure 9.2) would have all of the above components except a set of effectors; its performance would be judged by comparing its symbolic output to some desired "correct" classification. Most currently existing adaptive devices are adaptive classifiers. Now-common examples of adaptive classifiers are trainable, character recognition programs which take sets of primitive features (lines, strokes, pixels, enclosed regions, topologies, etc.) and classify them as discrete characters (e.g. ASCII characters). The performance measure in these programs is related to the proportion of "correct" versus "incorrect" classifications.

Another example are neural nets which are trained to distinguish the classes of "good-risks" and "bad-risks" in lending by using features such as financial status, type of investment, interest rates and other loan terms. The various possible combinations of feature values are classified into lend/don't lend categories. In this case the performance measure is the track record of the loans, the proportion of "defaults" in the group given loans by the net.

Adaptive classifiers also correspond to many of the model optimizations that go on in scientific and engineering models. Once the observables have been chosen, the task of predicting a given phenomenon often involves optimizing the various parameters to achieve better predictability. The particular model which comes about is not only determined by initial assumptions, but is contingent upon its performance.

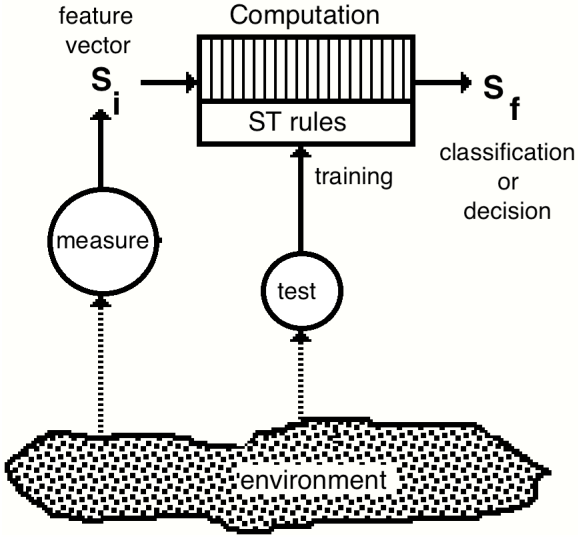

Figure 9.2 Schematic for an adaptive classifier. The device receives an input feature vector from the world, partitions the vector space through a computation, and is modified according to its performance.

Adaptive controllers. An adaptive controller (Figure 9.3), instead of having measured features, would instead have symbolic inputs, an adaptive computational part, and a set of effectors implementing actions. Performance would be judged by the effectiveness of the actions.

An example of an adaptive controller would be a numerically-controlled machine tool in which the quality of the product is monitored. The program which determines the motions of the effectors (e.g. grinders, millers, lathes, welders, etc.) is made contingent upon some measurable quality of the product (e.g. machine specifications and tolerances, durability, reliability).

Many technological processes of design and fabrication become such adaptive systems in themselves when the human beings who evaluate performance and make adjustments accordingly are included in the definition of the system. For the sake of clarity, however, we will consider only devices which themselves can carry out all the functions needed for syntactic adaptivity.

Another example is the Blind Watchmaker program of Richard Dawkins (Dawkins, 1987; Dewdney, 1988), which adaptively constructs graphical figures by evolutionarily presenting a set of graphical alternatives for a human user to select from, then mutating and mating them to generate a new generation of alternatives. The program employs a generative algorithm (*sans* linkage and crossover) to encode a graphical pattern language. An early version of this idea can be found in Gause and Rogers' use of a genetic algorithm to produce successive generations of graphical figures for human aesthetic selection (Gause & Rogers, 1976).

Contemporary examples of adaptive devices

In the biological realm, individual organisms capable of learning within their coordinative parts can be seen as adaptive devices to the extent that those parts can be cast in terms of syntactic, rule-governed relations. To the extent that discrete, rule-governed, percept-action mappings can be discerned and to the extent that these are modified by experience, most higher individual organisms can be seen as adaptive devices.

Within the realm of artificially constructed devices, there are many learning machines that are syntactically adaptive. Outside of the noncomputational functions they perform, the major differences between various adaptive devices lie in their performance dependency, in their training rules. These can encompass virtually all optimization techniques, from Monte Carlo random search techniques to statistical, Bayesian methods to completely determinate, gradient ascent mechanisms. Some of the adaptive digital devices developed in the 1950's and 1960's include Rosenblatt's perceptrons (Rosenblatt, 1958) , Selfridge's pandemonium model (Selfridge, 1958), Uttley Machines (1958) linear and nonlinear classifiers (Nilsson, 1965), Bayesian classifiers, and "evolutionary programming" classifiers (Fogel, Owens & Walsh, 1966).

In the 1950's many analog classifiers and controllers were developed along similar lines, with "open-loop" devices corresponding to the formal devices and "closed-loop" devices corresponding to adaptive ones. W. Ross Ashby's homeostat and Grey Walter's adaptive, cybernetic turtles were both conceived as analog, negative feedback devices. Ashby's homeostat was given the task of maintaining a constant output in the face of all sorts of input perturbations (Ashby, 1952, chap. 8). While this kind of adaptive "ultrastability" is quite useful for continuous, parametric inputs, it is less effective for nonparametric, discontinuous domains (p.119-120).

Major contemporary types of adaptive devices include adaptive neural nets (Grossberg, 1988; Hopfield, 1982; Hopfield & Tank, 1986; Edelman, 1987; Kohonen, 1988; Lippman, 1987) , genetic algorithms (Holland, 1976; Gause and Rogers, 1976; Goldberg, 1989; Wilson, 1986, 1987), and connectionist learning networks (Rumelhart*et al*, 1986), as well as relaxation algorithms and Boltzmann machines (Kirkpatrick et al, 1983, Ackly, 1985). In the historical development of artificial intelligence, funding for these learning devices was often pitted against funding for pre-programmed, formal-computational, non-adaptive strategies, leading to bitter rivalries (Dreyfus & Dreyfus, 1988).

Real and simulated adaptive devices

Many of the adaptive devices discussed so far are more often than not conceptualized and used within completely symbolic domains. Neural nets, connectionist architectures, and genetic algorithms are often first simulated on computers, resulting in a blurring between the simulation and the real world device. This reinforces the prevailing lack of any clear distinction in computer science between computers and robots, performance in a simulation vs. real world performance, and nonadaptive vs. adaptive behavior.

These distinctions must all be recognized before adaptive devices can be seen as qualitatively different from computers and fixed robots. In a computer simulation the semantic and pragmatic aspects of the simulated device are completely syntacticized; they become one part of the entire simulation. For adaptive devices to be seen as qualitatively distinct there must be a separation between the parts of the device that implement the performance measure and the other parts. This is generally true for real world devices, but not the case for simulations. The situation is similar historically to the situation before the concept of feedback was developed:

> In 1868, the brilliant physicist James Maxwell developed the first comprehensive theory of Watt's motion governor...; yet it is clear from his paper that he imagined the problem to concern the dynamics of motion, not the circuit of information. Restricted in this way, Maxwell's paper gave rise to the classical theory of speed regulation but not to a generic theory of feedback.
>
> The historical verdict is clear. A society committed to the mobilization and transmission of power could only see control devices as constraints on machine motion. In principle and design, the governor was considered an extension of cams, clutches, screws, and belts. Moreover, as we have argued, only metaphorically could one describe the governor as a "sensor," the valve as a "comparator," and the steady-state speed of the engine as the machine's "goal." We are led to one inescapable conclusion: feedback-based control devices become important only when certain fundamental problems of power generation and transmission have been resolved and when sensing instruments are developed that are physically and conceptually distinct from their machine infrastructure. (Hirschhorn, 1984, p.33)

In the case of the steam governor, the pragmatic comparator was not functionally differentiated from the other parts because it was not conceptually distinguished as a different kind of structure. One could say that a society committed to the mobilization and transmission of computed information can only see their devices in terms of computational symbol manipulations.

This is seen most dramatically in the framing of the parallel distributed processing (PDP) approach to machine learning (Rumelhart, Hinton & McClellan, 1986). Connectionists and the neural network researchers are the strongest contemporary advocates of adaptive devices, although there is a pervasive tendency to see adaptive devices as purely computational ones, albeit ones with specialized algorithmic structures. The parallel distributed processing perspective was developed as a means of integrating these bottom-up learning strategies. In the definition of PDP devices there is only symbolic manipulation:

> There are eight major aspects of a parallel distributed processing model:
> 1) A *set of processing units*
> 2) A *state of activation*
> 3) An *output function* for each unit
> 4) A *pattern of connectivity* among units
> 5) A *propagation rule* for propagating patterns of activities through the network of connectivities
> 6) An *activation rule* for combining inputs impinging on a unit to produce a new level of activation for the unit
> 7) A *learning rule* whereby patterns of connectivity are modified by experience
> 8) An *environment* within which the system must operate
> (Rumelhart, Hinton, & McClelland, 1986, p. 46)

It is apparent from the rest of the discussion surrounding this definition that "environment" here is referring to a symbolic environment and not to a real one. There is no explicit mention of a performance measure or even a real world environment. The problem of external semantics gets lost completely in the discussion of algorithmic structure: which output functions, connectivity patterns, propagation rules, activation rules, and learning rules to adopt.

What about adaptive devices which are embedded in symbolic environments or simulated? Are they still adaptive by the criteria developed here? This depends, as always, on the relationship of the observer to the device, on the observational frame, but generally such simulated devices will not appear to us to be adaptive ones. Once simulated within an artificial environment, the combined simulated-adaptive-device-plus-simulated-environment will appear to us as a formal-computational device. All processes will appear as computations. As such, the device will no longer appear as an adaptive device, since neither measurements, nor controls, nor performance measurements are carried out.

If all information regarding the simulated environment is dropped from the observer's purview, such that the simulated environment now appears to involve nonsymbolic interactions and the device-environment boundary now involves measurements, controls, and performance measures, then the simulated device will in fact appear as an adaptive one. However, we could not then claim that the entire simulation (device + environment) was an adaptive device. By truncating our observational frame in this way we have not gained anything, since all of the information that the device could produce was available to us in the first place. This kind of truncation does not advance our knowledge in the same way that an adaptive device operating in a genuinely unknown environment does.

This distinction is important because so many adaptive devices are first simulated in artificial environments, hindering distinctions between adaptation in a simulation and adaptation in a real world context. The pole balancing simulation of Barto, Sutton, and Anderson (1983) serves as a useful example. A pole rides atop a cart connected to the cart via a hinge. The cart is free to move in one dimension and impulses are given to the cart by an adaptive controller. The object of the controller is to keep the pole balanced on the cart by moving the cart appropriately, much as one would balance a pole on the end of one's finger. The state variables and parameters involved in the simulation are the position of the cart on the track, angle of the pole, cart velocity, angular velocity of the pole, pole length and mass, cart mass, force of gravity, the coefficients of friction between the cart/track and at the hinge, and the magnitude and allowable time intervals of the impulses. The requisite differential equations were solved using numerical methods. The point of the simulation, of course, is to test the adaptive characteristics of a particular type of neural net, to see if the neural net will produce the appropriate responses in a simulated environment. Can it learn to balance the simulated cart-pole system? As it happened, after roughly 50 trials the neural net controller began to keep the pole balanced for longer and longer intervals.

If we observe all the variable states connected with the cart, pole, controller, and environment along with their rules of interaction, we will have a state-determined system. This system will be expressible via deterministic state-transition rules, by computations. If we only observe the primitive features distinguishable by the controller, 3 cart position intervals, 6 angle position intervals, 3 angle velocity intervals and 3 cart velocity intervals, then the controller will appear to be adaptive within the artificial environment *because the environment no longer appears artificial*. It will no longer appear completely computational in character.

Pragmatically, we gain different sorts of knowledge from operating in real and artificial environments. *We always have complete access to a simulated artificial environment, while we never have complete access to real environments*. Consequently, we can exhaustively characterize the artificial environment by enumerating all the possible states of the simulation's state variables, we would be exploring the logical consequences of a closed set of mathematical assumptions. Remember, the point of the pole-balancing simulation was not to explore the artificial environment but to test the neural net.

Simulations are indispensable (conceptually and economically) in designing new devices, in terms of testing the logical consistency of the computational part, but, nevertheless, no information about the world outside the simulations is gained by running them. Computer simulations only enable us to test the logical consequences of our assumptions; they do not yield information about the world that was not already embedded in the design of the simulation.

In the real environment, however, we would use such an adaptive device to learn to perform a task for us, to find the best combination of the features and actions we have at our disposal. Here we face a situation where we do not have an exhaustive description of either the environment available to us, even in principle. A real implementation of such a pole-balancing device would yield information about how to deal with all of the physical factors involved in the real situation but left out of the simulation. Taken into account in the real world performance are properties like wear and tear, aerodynamic drag, nuances of motor response, air currents, magnetic fields, humidity, and all others, measured and unmeasured. An adaptive device operating in a real world context solves our problem for us directly, taking all of the real world factors into account when the quality of that performance is measured.

Most importantly, the two environments, real and artificial, are of different relevance to us as human beings because the problems we ourselves face as material beings all have to do with getting on with life in a real environment.

Characteristic behavior of adaptive devices

Since the state-transition functions of adaptive devices are ultimately contingent upon their effects in the external world, it matters little which particular parameter values for the state-transition function they start with, provided that their training procedures are convergent. If such a mapping is chosen at random and the system is allowed to run the performance of the system should generally increase as the state-transition function becomes optimized over the experience it has accumulated(figure 9.4).

Within each round of training this general rule may be violated. Training consists of exposing the device to interactions with its environment, measuring its performance, and changing its state-transition rule according to fixed training rules. Training is a testing process. A given state-transition rule may perform worse than its predecessor. The degree to which a device's performance can degrade, however, can be controlled by retaining the "best-so-far" input-output function and substituting it if performance is degraded too far.

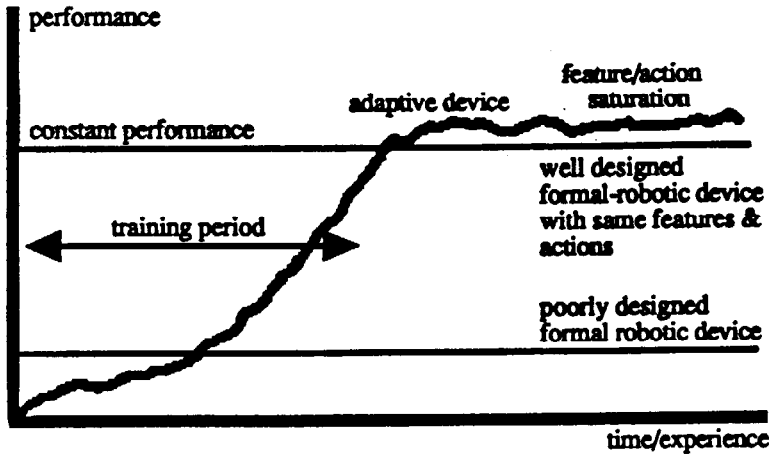

Figure 9.4 Characteristic performance of an adaptive device over time

These considerations are a major part of the art of designing adaptive devices and efficient optimizing techniques in the face of very complex "performance surfaces," having multiple local

maxima. If one examines the performance of these devices at intervals much larger than the testing/adjustment cycle, their performance will appear to be monotonically increasing.

The concept of "perfect performance" is most often seen within encoded problem domains, where formal, digital agreement with a standard is possible. An example would be the 100% "correct" recognition of characters by an adaptive classifier.

In analog domains "perfect performance" means that the device has achieved a performance level at or exceeding the resolution of the performance-measuring device. One could imagine an adaptive metal cutting machine which produced tolerances on the same order as those used to evaluate its performance. When perfect performance is reached adaptation stops; there is no further measurable improvement.

It may be the case, however, that the input features and output actions are not in themselves sufficient to achieve perfect performance. An example would be if one tried to classify characters in this typeface by means of the number of enclosed regions in each character. It would not be possible within such a feature space to distinguish between an "a" and an "o," but some distinctions could be correctly made. The performance of a device based on these features would saturate at some point, levelling off when the functional limits of its features and actions are reached. These are also the functional limits of its sensors and effectors. No better coordination between the two sets can be achieved at this point, and no better performance vis-a-vis the environment can be expected, given the hardware at hand. *An adaptive device can only be as good as the features and actions it has available to it.*

Syntactic closure

Currently, most of the work on adaptive devices is focused on developing efficient decision functions/ training rule combinations. A complex classification problem may require large numbers of training trials before showing appreciable improvements in performance. These problems are certainly real, but progress is being made and the general problem of optimizing large input-output functions does not appear to be insurmountable.

Adaptive devices can have as their input-output functions any function mapping the set of all perceptual distinctions onto the set of all possible actions. While this space of possible input-output functions can be quite large, it is always finite and expressible within a finite notational system.

> There are two defects in any [adaptive] control mechanism of the kind we have just considered in detail, which puts it out of court as a possibly organic decision making system. There is, first of all, the extremely practical question of how objective, utilities and rewards are defined, and the fact that such a machine may learn a decision function, only if (as noted later) a special type of reward is offered to it. Secondly, there is the involved issue of closure. Briefly, any such machine is finite, in the sense that its possible actions, and the events it may possibly recognize, form a closed set. (Pask, 1958a, pp.777-778)

In terms of perception-action mappings, adaptive devices are not completely replicable or predictable, since their behavior is dependent upon measurement and control interactions with the real world which cannot be completely predicted given the initial conditions of the device and its state transition rules. Such devices are, however, circumscribable in their behaviors, since the actual behavior of the device must fall within some enumerable combination of fixed percept and action possibilities. Even if we do not know the specific state-transition function, we can always circumscribe the set of all possible state-transition functions. The space of possible functions is still closed.

The limitations of fixed semantics

More fundamentally, inherent limitations of adaptive devices flow from their lack of plasticity in their semantic functions, that they themselves do not construct their relationships to the external world. The semantics of their internal states are determined by their sensors and effectors, which are stable physical structures. Adaptive machines consequently have fixed feature spaces and fixed action spaces.

While they can optimize their behavior within the categories given them by their sensors and effectors, such devices are ultimately limited by those categories.

In terms of biological evolution, adaptive devices are analogous to organisms incapable of modifying their senses to interpret the world in a different way, but nevertheless capable of optimizing their behavior within that fixed interpretive framework.

There has been remarkably little discussion in the adaptive systems literature of the limitations imposed on adaptive devices by their fixed perceptual and behavioral categories.

If the assumptions of the computationalists are that all "essential" aspects of cognitive behavior can be captured via computational symbol manipulations, the assumptions of nearly all the early advocates of adaptive systems, (e.g. Ashby, Uttley, Selfridge, Rosenblatt, von Foerster, et al) were that the essential mechanisms involved were adaptation within fixed feature and behavior spaces. Syntactics were what really mattered; they were the logic of the system; they were the domain-independent characteristics. The semantics, on the other hand, were non-logical; messy, domain-dependent, and anyway they were just a matter of finding the right peripherals for the job.

Part of the problem was that information theory, the interpretive paradigm of 1950's cybernetics, itself did not address the question of where the origins of particular encodings. Information theory was a theory about the transmission of already encoded information, but it was very weak in explaining the origins or meanings of bits of information.

If one examines the Homeostat, the Perceptron, the Uttley Machine, Pandemonium, or the Self-Organizing Systems of the late 1950's and the evolutionary programming of the early 1960's, it is immediately evident that they all operate within fixed categories. The failure of many of these devices to achieve their designers' overly optimistic claims, along with the rapid rise of computational power and better programming languages led the field of artificial intelligence to look to more computationally based solutions.

The early devices became the precursors of those being developed today: very roughly speaking, perceptrons and pandemonia became neural nets, evolutionary programming became genetic algorithms, self-organizing systems became Boltzmann machines.

There was very little discussion then of another level of adaptation which might be structural in nature, involving the categories within which "adaptive" mechanisms might operate. When these issues were discussed, they were discussed around the fringes of some other problem. In only one place in the literature has this author found any direct reference to the problem. In Gordon Pask's (1958b) paper "Organic analogues to the growth of a concept" there is a discussion of how new sensors and sensitivities might arise. We will discuss this paper in the next chapter.

Chapter 10. Evolutionary devices

> All the fundamental areas of physics, with the exception of the life-threatening high-temperature and atomic physics, have been opened up and utilized to advantage by living things. Almost all the basic principles of mechanics, dynamics, thermodynamics, optics, acoustics, had already been in the service of life for millions of years before the human mind learned to understand and master their functions. Mankind must (and somewhat belatedly) realize that, to some extent, our technology repeats development which life has successfully gone through. (Tributsch, 1982, p.204).

> The evolutionary history of the organism ... is embodied in the organism's activity. Therefore its knowledge is in its process; it is not something static, set aside from this process. Thus instead of a description of the organism's world emerging from the organism as subject, which is the Cartesian or Kantian way of looking at knowledge, we have the organism emerging from the world as an organized, coherent whole in which knowledge is a constituent activity: a constraining, ordering constituent.... (Goodwin, 1978, p.124)

In the last chapter a class of devices was outlined which modified their syntactic, computational parts contingent upon their performance in the external environment. It was concluded that these devices, however effective their learning algorithms, would always be prisoners of their semantic domains, of the fixed sensors and effectors which link them to the world.

This chapter outlines a class of devices which are capable of modifying their semantic relations by adaptively constructing their sensors and effectors. This is done through a performance-dependent feedback loop which selects the set of sensors and effectors to be constructed. Natural examples include the evolution of sense organs and body appendages as well as the operation of the immune system. Artificial examples include the fabrication of measuring instruments and tools of all sorts. One device, developed by Gordon Pask in the 1950's, which adaptively constructed new semantic relations unaided by human intervention is discussed. In the latter part of the chapter, the design principles involved in building an evolutionary device are discussed.

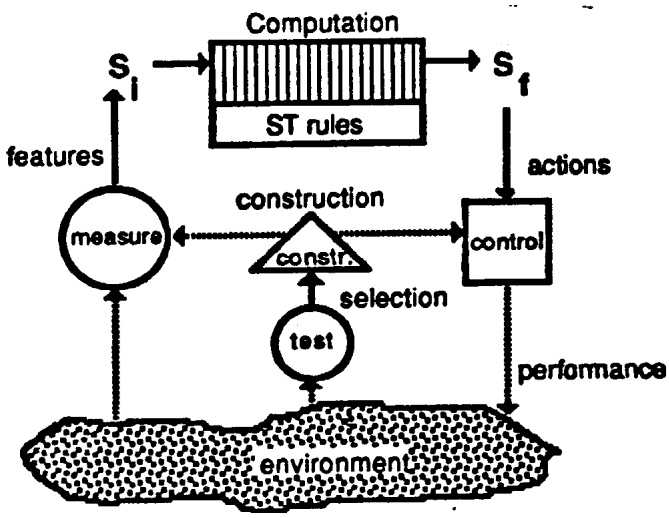

Figure 10.1. Structure-function relations for an evolutionary device with adaptive semantics but without adaptive syntax.

Definition of evolutionary devices: adaptive semantics

An *evolutionary device* is one whose semantics are adaptive. Evolutionary devices solve the problem of finding appropriate features and actions for a real world task. The sensors and effectors of such devices can be modified through experience so as to optimize the features and actions which are available to the device. In this way the input-output domain of the device is altered. The essential structure-function relationships of such a device are illustrated in figure 10.1.

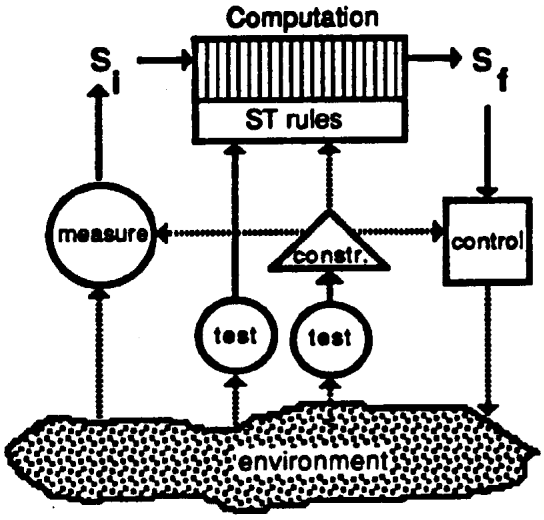

Figure 10.2. Structure-function relations for a general evolutionary device. Both syntactic and semantic functionalities are adaptive. Note that there are two levels of adaptation in the syntactic paper: adaptation by syntactic switching and adaptation by physical construction.

Evolutionary devices thus consist of a set of sensors, a set of effectors, a computational part, a performance measure, and an apparatus for constructing new sensors and effectors. The constructive part implements a control process, a genotype-phenotype relation, transforming symbols into physical structures.

A device which has both adaptive syntax and adaptive semantics, that is one which is both *evolutionary* and *adaptive*, will be called a *general evolutionary device*. The structure-function relations for such devices are shown in figure 10.2. Having adaptive syntax in addition to adaptive semantics greatly facilitates the coordination between the two domains. In addition, the construction process of the general evolutionary device also directs the formation of the physical structures necessary for the computational part. New computational primitives in the form of new computational states can thus be constructed by these devices. The syntactic part of these devices is thus expandable, but not infinitely so. The possible number of computational states depends crucially on the nature and limitations inherent in a given construction system.

The observed behavior of an evolutionary device will involve the measurement-computation-control state transitions sequences of formal-robotic devices as well as performance measurement-construction sequences leading to changes in the measurement and control state-transitions. These changes will be apparent in measurement and control state transitions either through shifting probabilities of existing state-transitions or through the appearance of new state-transitions that were not observed before and/or the disappearance of old state transitions. If these changes are functionally significant, they will also exhibit changes in the measured performance of the device. Thus we would observe a performance measurement leading to changes in measurement and/or control transitions leading to changes in subsequent performance levels.

Construction

What training rules are for the syntaxes of adaptive devices, the constructive parts are for the semantics of evolutionary devices. The major difference is that the training rules of an adaptive device are themselves computations, whereas the construction apparatus of an evolutionary device has the formation of a sensor or effector as its result. One is a symbolic switching, while the other is a symbolic constraining of a nonsymbolic process.

This is clearest if we consider in the general evolutionary device both the symbolic modification of the computation performed (as in adaptive devices) and the construction of the computational part itself. The training rules of the adaptive device switch between computations which the device is already structurally capable of performing. This is similar to running various programs on a microcomputer having a certain limited amount of RAM; which programs are run can be switched symbolically, without

creating new states or altering the structure of the machine. On the other hand, the construction process would be similar to the method by which the machine was fabricated, and in this process it would be possible to add more memory or to add new symbol primitives to the machine's structure.

The difference between symbolic switching and physical construction processes has implications regarding the open or closed nature of the behavioral repertoires of evolutionary devices. We will discuss these implications in detail later in this chapter.

With the addition of constructive processes into general evolutionary devices, we now have three time scales corresponding to the three device-environment feedback loops. We have the ongoing measurement-computation-control-environmental effects cycle, which, biologically speaking, corresponds to physiological time. We have the slower, syntactic adaptation loop, which corresponds on a population level to microevolutionary optimization of gene combinations within a fixed gene pool, and on an individual level to learning within fixed primitive categories. Finally, we have an even slower, semantic adaptation loop, which corresponds to qualitative structural evolutionary changes over phylogenetic time periods, and which corresponds on the individual level to the formation of new conceptual categories. In general evolutionary devices, there is also the adaptive construction of new computational parts, enabling more states and/or more mappings (greater flexibility). This process would take place on the same time scale as semantic adaptation, dependent as it is on the construction and testing of new physical structures rather than the switching from one already existing coordinative mapping to another.

The biological evolution of new semantic relations

The most obvious examples of evolutionary devices come from the biological evolution of new sensory organs and body structures. Biological populations over phylogenetic time periods solve the encoding/decoding problem by constructing their own sensors and effectors. Over the course of evolution those sensors and effectors which provide the cognitive and behavioral alternatives conducive to survival will be propagated via hereditary mechanisms. The construction of sensors and effectors and their associated categories of perception and action are thus part of an epistemological feedback loop running through the environment and the construction processes themselves. Through this cognitive self-construction the organism acquires a degree of epistemological autonomy (Maturana, 198), of semantic closure (Pattee, 1985). As a result we see the optimization of sensors and effectors over phylogenetic time spans, creating new primitive percepts and actions in the process.

To the extent that biological evolution also encompasses the evolution of behavioral plasticity and ontogenetic learning mechanisms, biological evolution furnishes examples of general evolutionary devices.

The immune system as an evolutionary device

The immune system is a concrete, natural example of a device which constructs its own sensors, thereby constructing the very categories by which it apprehends the world around it. On the most basic level, antibodies are constructed and differentially replicated on the basis of their ability to bind antigens. Each antibody interacts with various substrates in different ways, each implementing a different measurement, each measurement enabling the immune system to make another perceptual distinction. The perceptual category which results is the presence or absence in the immune system's environment of any antigens which can bind to the given antibody. The various recombinative and hypermutational mechanisms in the immune system allow for a large variety of possible antibody-sensors and an efficient search of these possibilities.

Immunity is generally characterized by high affinity antibodies. Studies performed on whole serum antibodies ... have implicated the role of antigen as the selective agent in the affinity maturation process. Our results provide a

> molecular description of this phenomenon. Antigen selects not only from the diverse pool of unmutated V[ariable] region structures expressed by the preimmune B-cell population but also from the additional diversity apparently generated by stochastic mutation of these V regions, resulting in a highly adaptive process in which many structural solutions to a common functional problem (i.e. achieving high affinity) are reached. From a finite preimmune repertoire, less than or equal to the number of B cells in the host, functional antibodies of seemingly infinite variability are thus generated. (Wysocki et al, 1986).

The immune system thus optimizes its perceptual categories in much the same way that cognitive and behavioral categories are optimized during morphological evolution, through mutation and construction. Because the rest of the immune system does not distinguish which particular antibody is recognizing an antigen, the mapping of a particular antibody's recognition to the subsequent immune response is fixed. The immune system thus has an adaptive semantic part (the antibodies) and a nonadaptive, fixed syntactic part (the rest of the immune response). The immune system therefore is only an evolutionary device, not a general evolutionary one.

The construction of new artificial sensors and effectors

Scientific communities can be considered to be evolutionary devices, when they construct new measuring devices, creating new independent observables and potentially, new regimes of predictiveness. The activity of constructing new measuring devices external to the body and selecting those which seem to enhance some particular human endeavor also has similarities to biologically based sensory evolution. When new sensors are constructed, new distinctions become possible, enabling new percept-action contingencies. The construction of devices such as chronometers, thermometers, microscopes, telescopes, cloud chambers, spectroscopes, astrolabes, seismometers, and geiger counters, allows us to make new distinctions on the world that we otherwise could not make, thereby allowing us to act contingent upon their measurements.

Likewise, technological evolution has its parallels to the evolution of new effectors in the evolution of biological appendages. External devices potentiate functions that we already have as organisms: mobility is drastically increased by transportation vehicles, hunting and gathering of food become augmented by agriculture, face-to-face communication becomes dramatically extended in time and space through telephones and preserved text, and so on.

As their purposes must serve biological organisms, our technologies in one sense or another amplify biological functions already existing in our bodies. They become extensions of our bodies, new prostheses which we integrate and absorb as our own.

> Our subsidiary awareness of tools and probes can be regarded now as the act of making them form a part of our own body. The way we use a hammer or a blind man uses his stick, shows in fact that in both cases we shift outwards the point at which we make contact with the things that we observe as objects outside ourselves. While we rely on a tool or a probe, these are not handled as external objects. We may test the tool for its effectiveness or the probe for its suitability, e.g. in discovering the hidden details of a cavity, but the tool and the probe can never lie outside the field of these operations; they remain necessarily on our side of it, forming part of ourselves, the operating persons. We pour ourselves out to them and assimilate them as parts of our existence. We accept them existentially by dwelling in them. (Polyani, 1964, p.59)

When we incorporate a tool into our bodies we change our relationship to the world at large as if we had evolved a new body part, thus creating a new semantic relation with the world. We notice this process most when we are in the transition phase of establishing the automatic linkages needed to incorporate the tool into our bodies.

> In a recent paper, Bullinger [1981] has developed Piaget's sensorimotor theory starting from a clarification of the distinction between the subject and the subject's body. Not only during early sensorimotor coordination in infancy, but throughout life, the body is "cognitively transparent" only when the subject is not confronted by new tasks. A new task

brings out new, objective properties of the body--physical and informational properties which were always latent within it as a complex, organized, biological object, but which never had to be internalized until the moment when the new task or situation made them evident. Learning to use a pencil throws up new material (physical and informational) properties of the hand that were not known or knowable through any previous experience, and until you internalize these you cannot use a pencil as an extension of your body. (Churcher, 1982, p. 298; Bullinger, A., 1981, Cognitive Elaboration of Sensorimotor Behavior, in: "Infancy and Epistemology," G. Butterworth, ed., Harvester Press, Brighton.)

In general, tools are constructed to implement new, stable relations with the world, making them amenable to control. By constructing many alternative fixed sensory prostheses and tools we are able to enlarge our perceptual and behavioral repertoires. The combination of human beings plus the constructional system for fabricating new sensors and effectors is thus an evolutionary device.

In order for us to utilize these other windows on the world, we must construct linkage relations between our natural senses and the new measuring devices and controls (Rosen, 1978, chap. 4), in effect incorporating the measuring device into ourselves. When we construct these linkages, we push the boundary of semantic relations outwards by establishing a syntactic linkage with the symbolic output of the measuring device.

Consider the case of a robot equipped with a television camera which is reading a household water consumption meter. Assume the meter is of the digital type. When the robot reads off the numerical value of the meter with its camera, it transforms the digital output of the meter to features in its visual array which will then get reclassified as symbols. If the robot is performing effectively, this will be a deterministic, one-to-one mapping between the output symbols of the meter and the symbols resulting inside the robot from the classification of the video features. This will be describable by a syntactic transformation between the meter symbols and the internal symbols of the robot. The syntactic realm has been extended outwards so that the water meter effectively becomes another sensor, *as long as the robot is looking at the meter and recognizes that it is utilizing a different sensor than just an unmediated video picture*. This recognition can be accomplished through syntactic adaptation. (How do we know to interpret what we see through a telescope differently from how we see normally? We associate the features of the telescope's appearance with the changed perceptual-semantic relation.)

The semantic realm also gets pushed outwards. In the general robot-camera situation, the symbolic-nonsymbolic boundary lies within the camera: distal to the camera is an unencoded, nonsymbolic realm, proximal to the camera is an encoded, symbolic one. When the water meter is connected, and a syntactic mapping of the video features of the meter to the internal symbols of the robot is achieved, the boundary jumps outward to the water meter. On one side of the meter is an unencoded, unpredictable water flow, on the other side a digital readout.

This is related to Bohr's statement that the dividing line between the observer and that which is observed is not absolutely fixed and that it is under the control of the observer. The observer chooses which measurements are made by choosing between different available measuring devices and also by selecting the interpretation of what s/he is sensing. (Am I sensing through the water meter or am I seeing through my own eyes?). Here we have chosen to put the device/subject boundary at the point where we as external observers see measurement and control observed state transitions rather than at some fixed point. The result is the same. Any number of external devices can be constructed and used to generate observables on the phenomenon, such that the boundary of the symbolically constituted subject is continually shifting relative to particular physical structures.

Functionally similar to the construction and selection of sets of external, fixed sensors and effectors would be the construction of adaptive prostheses.

From this perspective, there is a functional equivalence between parts of your body, artificial sensory prostheses, and tools: a blind person's stick, an aeroplane (for the pilot), a mechanical digger (for its operator), the 'sonic guide' that a few blind babies have now worn, the tactile stimulator attached to the skin and connected to a video camera on

the head (Bach-y-Rita, 1972). But such artefacts also have in common the fact that although users' internalization of them may change and develop, they themselves do not. A stick, for example, does not adapt when you use it for a probe, it doesn't change its informational properties (even though new situations may arise which reveal hitherto unused properties). Nor does the camera on your head change its way of transferring information to the stimulator on your skin.

But what will happen when the prosthesis is adaptive? (Churcher, 1982, pp.298-299; Bach-y-Rita, P., 1972, "Sensory Substitution", Academic Press, New York.)

Organic analogues to the growth of a concept

How would we build such an adaptive prosthetic device? Churcher points us in the direction of Gordon Pask. As far as I know, Gordon Pask is the only person to explicitly envision a device which creates its own observables and controls.[1] His papers of 1958-1961 (Pask, 1958a, 1958b, 1959, 1961) outline a series of experiments with platinum electrodes in a ferrous sulphate solution in which thread structures grow outwards to optimize the conduction of current:

> The energetic conditions which have been described are applied to this assemblage and in order to realize these conditions in practice a learning machine must be introduced such that it distributes the limited current available in a manner which ultimately develops threads. The most general learning machine is a device able to pass current through each of a finite set of electrodes, by making connection with these electrodes. Connections are made and are regarded as trials. The result of a trial may lead to greater or less current passing via the electrode to which it refers. The learning machine is programmed to learn that trial making activity which maximizes the current passed via each electrode subject to the limitation that only so much current may pass in the assemblage as a whole. (Pask, 1958b, p.917)

Each electrode represents a variable, and the connections between electrodes represent continuous mappings between the variables. The connections are established by the growth of the iron thread structures, which determine the connectivity between the electrodes. Connectivity between two variables is related to the amount of current passing between their respective electrodes. Various connectivities can be rewarded by passing more current through them.

> As described the rewarding procedure acts by supplying more current for constructing threads whenever the mode of problem solution, implied by the existence of a certain thread structure, satisfies an external criterion, such as maximizing the output of the process. In this learning by reward procedure some threads flourish, others will prove abortive. It is a lengthy and inefficient kind of learning not unlike natural selection. (Pask, 1958b, p.920)

Roughly speaking, we thus have primitive, distant analog relatives of a simulated annealing process or a Boltzmann machine (Kirkpatrick *et al.*, 1983, Ackly, 1985), "organic analogues to the growth of a concept" (Pask, 1958b). So far, if we only see this device in terms of relations between its state variables, voltages at the electrodes, the device has the function of an analog computer. If we discretize the input and output voltages such that the device shows deterministic input-output mappings, then we have a digital computer, a formal-computational device.

---

[1] We do, of course, have robots with tunable sensors and/or effectors (e.g. the artificial retina of Mead & Mahowald, 1988; Selfridge, 1982), but these are conceptualized as the optimization of existing measurement and control parameters rather than the creation of new ones. Because many of them are analog mechanisms, these devices lie in a gray area with respect to the taxonomy here (when does a tuning perturbation in a sensor result in a new sensor?). With purely analog devices, it becomes difficult to distinguish between measurements, controls, and computations and hence to talk about syntactic vs. semantic adaptivity. For robots with tunable sensors and effectors, like the mechanisms of biological organisms which are still poorly understood, we must first define an observational frame, observe their behavior, and determine if there is a distinguishable difference in their measurement and control behavior as a result of experience.

However, this device is potentially capable of other types of functions than computations. Suitably rewarded, it is also capable of making measurements and controls.

> We have, up to the moment, thought of the assemblage as connected to the external world by electrical channels, and it will be convenient to regard the connection from the assemblage into the external world as still mediated in this manner. The input connection may, however, be of any kind. True there will be some electrical connections established at the outset, but it is not impossible that changes of temperature, chemical constitution, vibrations, magnetic fields, and so on, will affect the development of the assemblage and serve as inputs. In general one of these arbitrary disturbances will be an input, if it is relevant on some logical ground, to the process, and if, in particular, the assemblage would be rewarded if it were able to sense the input in question. Thus, for example, a buzzing sound emitted by the process, although not explicitly included in the original set of inputs, as it might have been by provision of a microphone and an appropriate wire, will be relevant if reception of the buzzing signal without a microphone, leads to a state of the assemblage which would not otherwise have occurred, and which in turn gives rise to a rewarded state of the process.
>
> As a receiver of vibration, and thermal stimuli, a structure of threads is very inefficient. However, like any other physical system, it is to a certain extent sensitive to these variables and to the fact that its parts have no initially specified function, no deliberate attempt is made to minimize the effect of such disturbances on its workings. The characteristic which was demonstrated is that, supposing an arbitrary disturbance is rewarded persistently over an interval, and supposing that the assemblage is able to sense the disturbance occasionally, in the way that a computing machine might sense a vibration via a microphonic valve, the learning process will lead to a region of the thread structure specifically adapted to reception of this disturbance, so that when it is repeated it exerts an increasing effect upon the state of the system. Put crudely some region of the thread structure becomes adapted to react like a microphone....
>
> There is no sense in which a microphone has been designed into the system. Rather, as a result of learning, the system has selected attributes of the external world which are relevant to its problem solving activity but which were not originally specified as part of its input set. (Pask, 1958b, pp. 921-922)

As Stafford Beer tells it, Pask's vat with the platinum electrodes and ferrous sulphate solution actually did unexpectedly begin to amplify ambient sounds. They started hearing sounds from down the street (Beer, 1984).

A number of lessons are to be learned from Pask's experiment:

1. All systems are weakly totipotent with respect to sensitivity to the various attributes of the external world, but each system has different sensitivities to different types of perturbations.
2. The design of a device determines what the sensitive degrees of freedom of the parts are. If "no deliberate attempt is made to minimize the effect of such disturbances," and, in fact, sensitivity to disturbances is rewarded, then new measurements and controls can arise.
3. This causes us to broaden our conception of what connectionist devices might do, *as long as there are analog as well as digital processes going on inside them.* It should be noted that formal-computational simulations of these devices will not generate new measurements, because in a simulation, all relevant degrees of freedom of the simulated device "must be specified as part of its input set." *These networks must be built, not simulated.*

Up to now connectionists have been trying to perform computational tasks, to compete in formal realms. "The 'radical departure' of connectionism may best be thought of, likewise, as a change in some relatively basic ideas *within* the general computationalist approach" (Boden, 1988, p.252). Connectionist, "bottom-up," associationist approaches may not be able to compete with "classical," directly encoded symbol manipulations in terms of processing efficiency, but there are other functions that large mixed digital-analog networks can perform which classical symbol systems cannot, namely measurements and controls. We must break out of the computationalist paradigm to see connectionist devices in a new light.

This is reminiscent of the exchange after the "organic growth of a concept" presentation between Pask and Alan MacKay, who had been investigating similar electrolytic systems as "kinds of probabilistic storage devices" (Pask, 1958b, p.926). Pask says,

> For use in a conditional probability machine, I imagine, this sort of dependence, which Dr. MacKay has illustrated by referring to mechanical sensitivity, will be an embarrassment. However, I wish to emphasize that these very properties are those which, in the present application, we should try to encourage, in other words, I would like the threads to exhibit sensitivity to all sorts of disturbances, not only the particular set of disturbances which as a computer designer I might regard as relevant; for example, the set of disturbances which amount to changes in field parameters. (Pask, 1958b, p. 926)

4. Unlike a formal-computational device, where we can enumerate all possible input-output mappings, we cannot enumerate or circumscribe the set of all measurements or controls which are potentially accessible by such a device. Unlike syntactic spaces, semantic spaces are undefined, and this is what gives the expansion of measurements and controls its open-ended character.
5. The main limitations of this device are not in the closed nature of its potential behaviors, but in that the physical search is of a "lengthy and inefficient" kind. The tedious length of the physical search probably cannot be avoided entirely, but once a functional structure is found, it should be possible to re-create that structure without having to go through the physical search process each time. Here is where a symbolically constrained physical search can reduce the "recapitulation time" of the original search. This is the only way that an "information amplification," in connection with Ashby's discussion of Descartes' dictum (Ashby, 1952b; chap. 8), can result.

Tied up with issues of "constructional efficiency" are issues of "constructional reliability." It is not clear from Gordon Pask's paper how reproducible the strand structures and functions were. Perhaps by replicating the sequence of voltage changes of the training period, threads with at least approximately the same emergent properties could be grown. We would want a construction process which was open-ended but not so unstable that useful structures could not be replicated.

Thus we arrive at some essential criteria for the design of an effective symbolically constrained construction system.

Desirable characteristics for effective construction

Heritability and replicability of results. For the organism or device to be able to provide sensors and effectors which can be reproduced in subsequent generations or in other devices, there must be a reliable means of transmitting the initial conditions and constraints for such a constructional process such that the results can be replicated.

Constructional reliability. Replicability also requires that the dynamics of the physical system being constructed be sufficiently constrainable that, given adequate initial state preparation and the appropriate constraints, the physical system will reliably end up in a stable functional state. Constructional reliability is necessary if the subsequent generations of organisms or devices are to achieve levels of performance similar to their predecessors.

Richness of structural alternatives. In order for the constructional system to be functionally useful, the system must access structures and functions not already attainable by the organism or device.

Richness of constructional alternatives gives the device more options. The more options, the more likely that some effective option is within the repertoire of the organism or device.

The advantages of constructional richness are counterbalanced by overhead costs. There is a trade-off between the costs of maintaining structural complexity and the usefulness of the additional functions that the complexity enables.

Open-ended vs. closed construction

Construction must be directed symbolically in order to achieve high degrees of heritability, constructional reliability, and high potential structural complexity. The discreteness of symbols allows for arbitrarily high accuracy in the hereditary transmission of constructional information, allowing for the replicability of successful constructed structures. But how can a process which is symbolically

constrained, give rise to more possibilities than are expressible in the notation of the symbolic constraint? How can the symbolically-constrained construction system escape the notational closure of formal systems? The answer lies in the fundamentally different relation between symbols and dynamics in the processes of computation and construction. These different relations are at the heart of why biological systems are open-ended while formal-computational systems are not.

Open-ended structure creation lies at the root of open-ended function creation. The measuring and control structures needed to realize these new functionalities must be constructed in a way which allows for both replicability of the new functions and open-ended creation of still more functions.

Protein folding as open-ended construction

Protein folding may be regarded as an archetype for the relation between symbols and dynamics in biological systems and as a prototype for an effective construction system. The advantage of considering protein folding is in its relative concreteness and the relatively advanced state of contemporary knowledge of the process. Several properties of this construction system stand out.

Dimensionality. A linear string of nucleotides determines an amino acid primary sequence which folds up into a three dimensional functional enzyme. Thus, the final protein structure which is formed is of higher dimensionality than the string which codes for it. There are more degrees of freedom in the protein folding process than in the DNA representation, allowing for one-to-many mappings of DNA strings to physical structures. A given protein in solution, far from having the single, static structure of its crystalized form, cycles through many different conformations. Each of these conformations interacts with the various molecules encountered in a slightly different way. If more than one three dimensional conformation is produced with one gene, then the population of conformations produced may have multiple effects. This is quite different from a computational process, where one symbolic state encodes one physical state.

Implicate, analog dynamics. Protein folding must rely on implicate, analog dynamics of the amino acid residues and the solvent to fold itself. Steric, hydrophobic, electrostatic, noncovalent and covalent interactions between the residues as well as an untold number of residue-solvent interactions constrain the folding process. This does not appear to be anything like a rule-governed process, and the kinetics of protein folding indicate that the process is neither totally directed nor totally random. Relaxation processes like protein folding and other annealing processes follow physical laws, not rules imposed upon the dynamics of the system through the construction of elaborate sets of programmable constraints. In this view, programmed, simulated annealing is qualitatively different from physical annealing. The digital representation cannot fully replicate all the properties of its analog counterpart. One relies on explicitly constructed physical constraints, while the other utilizes the inherent, autonomous dynamics of the physical system. Those physical systems which do use the inherent dynamics of matter become effectively nonprogrammable because the interactions are not precisely describable (Conrad, 1972, 1974, 1983). Their dynamics are implicate, irreducibly analog.

Variable degrees of freedom. Protein folding also may incorporate variable degrees of constraint in the construction process. Some protein chains may fold into more highly constrained tertiary structures than others. Less highly constrained proteins may cycle through many more thermodynamically less stable conformations. The degree of the one-to-many mapping between DNA strings and protein conformations can be modified according to the competing necessities for adaptedness and adaptability of function. This is similar to the "temperature" control in simulated annealing learning algorithms. Degrees of freedom are gradually frozen out of the construction system when adaptiveness and determinacy are needed. Degrees of freedom are added to the constructional process when adaptability is required. When new functions are needed, there may be selection for greater adaptability of proteins with lower specific activity with any one substrate, but broader effects (Pattee, 1973). When the specific activity of an enzyme needs to be increased to increase the speed of the reaction and/or the efficiency of

the enzyme, more constructional determinacy may be evolutionarily favored. The amino acid sequences selected for would fold up into fewer, more stable, native conformations with greater specific activity towards their primary substrates, but a narrower range of effects on other substrates.

Analogous reductions in degrees of freedom occur during the evolution of the immune response, as recombinatory sequence rearrangement mechanisms are succeeded by hypermutation point mutation mechanisms. Genetic recombination allows for wide variation, while point mutations allow for finer optimization of the antibody structure. The result is a kind of "generalist-specialist mechanism" (Weinard & Conrad, 1987).

Chaotic generation processes have been suggested as possible means for supplying the necessary variability for evolutionary mutational processes to function (Conrad, 1986). Beyond merely supplying "noise" to biological systems, chaotic mechanisms may make it easier to control the *degree* of (apparent) stochasticity in the system. It has been speculated that chaotic processes in the brain might similarly regulate the degree of "random" variation in search processes there (e.g. Skarda & Freeman, 1987).

Principle of Function Change. Related to optimization of the degree of constructional constraint is the multiplicity of functions even one determinate conformation can have. The principle of function change asserts roughly that many different biological functions coexist at any time in any particular anatomical or biological structure, and that evolutionary development has the effect of shifting the weight assigned to one function carried out by a structure at the expense of the others (Rosen, 1973). Every protein affects the rate of every reaction to some degree, even though most of these interactions are too weak to be measured. Weak interactions between proteins may be crucial to the evolution of new function. There is some evidence that proteins may evolve from less stable, less specific forms with many weak interactions to more highly constrained, more specific forms with fewer, but stronger interactions (Kacser & Beeby, 1984). In formal computations either there is an effect or there is not, due to the discrete nature of the categories and their interactions. Weak interactions and field effects are not possible on the level of the primitive symbols in a computational domain.

Possibilities and limitations of evolutionary devices

Like formal and adaptive devices, the designer of an evolutionary device is faced with a specification problem, but in the case of the evolutionary device, s/he does not have to come up with specific sensors and effectors. Rather, some construction system must be specified through which sensors and effectors will be built. This might seem like more effort than attempting plausible features and actions one by one, until it is remembered that the situations in which one would construct an evolutionary device are those in which one has no plausible specific candidates for what will work. The construction of an evolutionary device is thus mandated only in the most intractable problems, when the essential categories for effective problem solution are unknown. The selection of the physical substrate for the construction system is therefore a heuristic search, guided by physical principles and intuition. Once a physical construction system is set up, however, many different problems can be attacked, much in the same way that the immune system, once evolved, can cope with an enormous number of potential pathogens.

The major limitation of evolutionary devices is the time needed to conduct the structural searches needed to arrive at useful measurement and control innovations. The measurement of real world performance, too, takes more time than its computer simulated counterpart.

Contemporary technologies of symbol manipulation are far ahead of symbolically directed fabrication of structures. At the present time it is more expensive to build structures than to manipulate symbols, so much more deliberate *design* by human beings is generally employed when nonsymbolic, physical action is required.

Nevertheless, even with the longer times to build sensors and effectors some of the examples given above can be implemented on quite reasonable scales. The large scale "systematic" search for antibiotics by pharmaceutical companies in some ways is similar to the action of the immune system, except that a construction language is not employed to construct the alternatives. This process could be automated and integrated with computer models for designing desired antibody conformations. Here the computer would be acting as part of the construciton system in the construction-performance-selection loop of an evolutionary device. The computer model would direct the construction of chains of amino acids (or whatever other primitive polymer subunits were used), the antibody would be tested, and the results would be fed back into the computer modifying the construction program. This is not so far away from amino acid analyzers, DNA sequencers, automatic chemical synthesizers and *in vitro* transcription-translation systems currently in use. It is not hard to envision using such an artificial immune system for the discovery of new antibiotics or as a means of differentiating and/or separating different but highly related biomolecules by their affinity for various related antibodies. Antibodies could also be adaptively produced to act as enzymes, either to attack specific pathogens or to carry out other chemical tasks (e.g. digest petrochemical wastes).

In the introduction, the possibility of designing such an artificial immune system for use in deep space probes of the future was raised. Here time is not so crucial as in terrestrial applications. A construction system fabricates polymers of various sorts (proteins or otherwise), the polymers fold-up and interact with various macromolecules they encounter in the immediate environment of the probe. When a folded form binds, it would change conformation, sending a signal (via other molecular detectors) to the construction system to produce more of that particular polymer and closely related variants. The signal acts as both a recognition that an antigen has been recognized and a feedback mechanism for increasing the sensitivity of the array of antibody sensors. Each different antibody has slightly different specificity, so adding antibody variants increases the dimensionality of the feature space. The whole gamut of mechanisms of the immune system and biological evolution could be used: recombination and permutation of stable subunits, genetic linkages, crossover, point mutations, various compartments implementing relatively isolated gene pools, modification of the chemical environment in which polymer folding takes place, and self-modification of the construction system itself.

Related possibilities lie in the realm of other bio-synthetic construction techniques, constructed reaction networks and "nanotechnologies." (Drexler, 1986, 1989). Once the techniques for automatic construction of symbolically-specified individual macro-molecules become possible, then the usefulness of automated structural searches for new features and actions will be more feasible, because the time scale for construction and testing will dramatically decrease. We will need to find ways of interfacing this microscopic world with our own macroscopic one, perhaps through reaction-cascades and other molecular amplification processes. We would need to consider what kinds of reaction networks could support what kinds of higher level functions (Minch, 1988), what kinds could act as amplifiers. Reaction networks could also form the basis of new structural primitives for higher level functions. In such a scheme, new, slowly changing primitives could arise out of networks of much faster underlying micro-processes (Minch, 1988, Section 7.4). With the development of nanotechnologies and various mixed digital-analog chips, physical searches could conceivably be carried out on very small time scales. If such a situation comes to pass, it may be more economical to optimize features and actions and to couple them directly together into "smart machines" than to go through computational adaptation.

Yet another area would involve designing human-machine interfaces so as to encourage as much as possible evolutionary interactions. The computer could function as an adaptive prosthesis for the human or vice versa. The development of flexible tools, ones which adaptively change their characteristics in response to their user, would also be possible (Hirschhorn, 1984). One could also imagine trainable, adaptive optical sensors for the blind, which could be adaptively constructed to detect relevant features for a particular context.

Human beings as evolutionary devices

Given the appropriate observational frame, humans with measuring devices, tools, and language can be seen as evolutionary devices. But what about humans without these external additions? Are there internal mechanisms for constructing new semantic relations?

Human beings, with all their biological and behavioral complexity, pose special problems of classification, because, unlike devices, there is no obvious appropriate observational frame. We do not as of yet have a very good idea of what the fundamental states of human beings are. We can readily identify measurement, computational and control operations in artificial devices because we have access to their internal states, but this is manifestly not the case for human beings.

Nevertheless, we can identify perceptual and motor repertoires in human beings which change over time contingent upon experience. One has only to think of the highly developed ear of a practiced musician or the fine motor control of a trained athlete to see that qualitatively new sensory discriminations and behavioral repertoires can come into being as a result of experience.

Less obvious is the creation of new concepts and their effects on how we view and act on the world. Such concepts can be viewed as *de novo* constructions which alter our basic perceptions of the world or combinations of existing perceptual primitives. This debate will only be resolved when more is known about the neurological processes involved. When these processes are known well enough to determine whether they are computational or implemented via measurements and/or controls, then more general observations regarding the type of adaptivity in various neural structures could be made. For example, neural assemblies have often been thought of as purely computational elements, formal ones like McCulloch-Pitts neurons or syntactically-adaptive ones like artificial neural nets. They could alternatively be considered as general evolutionary devices in their own right, making measurements, performing computations, executing control processes, all adaptively contingent upon their performance. We can speculate that, rather than being primarily syntactic elements, neurons may be primarily semantic in character. Neurons might be better thought of as elaborate measuring devices rather than computational elements.

Chapter 11 The limits of adaptivity

Once the device types are defined by their structure-function relations, it is possible to compare the different types. Different types of structural plasticity lead to different types of adaptivity, and consequently, to different capacities and limitations, strengths and weaknesses.

Adaptivity and improved performance over time

Given their internal structure, the typical performance characteristics over time for the various device types operating in stable environments with stable performance measures can be analyzed, as in Gause & Rogers (1981). These are plotted together in figure 11.1 on the next page.

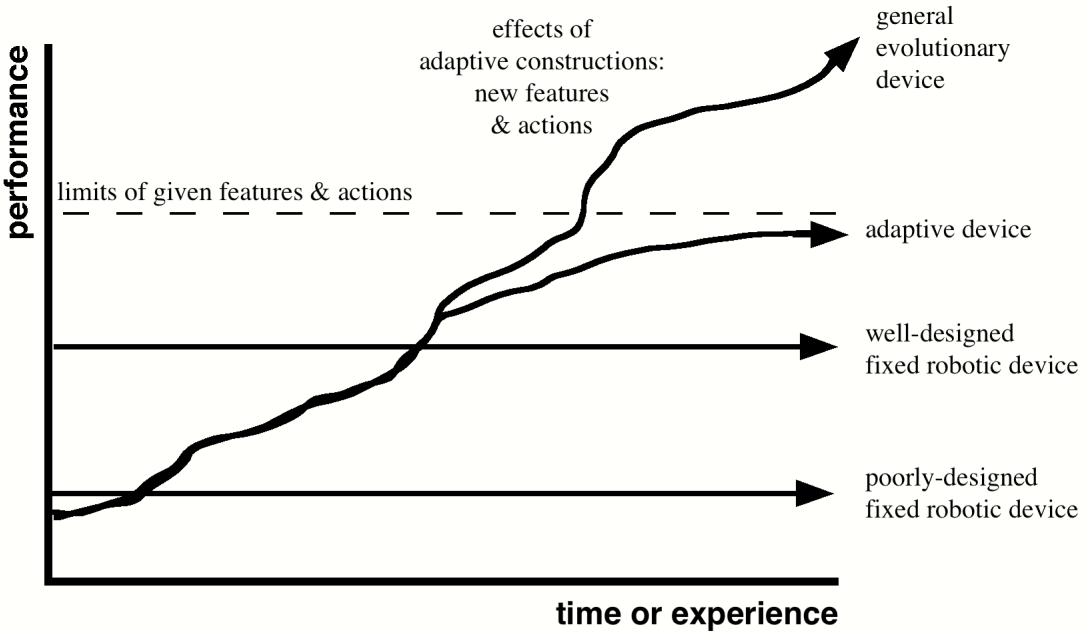

Figure 11.1. Summary of general timecourse of performance over time for different device types. Within this timecourse would be a great many iterations of measure-compute-control loops, somewhat fewer adaptive syntactic loops, and many fewer adaptive semantic loops.

Formal-computational devices do not by themselves have real world performances, since their outputs, their effects are entirely symbolic, and must therefore be mediated by human beings or sensors and effectors standing outside the device.

Formal-robotic devices have constant performance over time because all semantic and syntactic are stable.

Adaptive devices will show general improvement over their training periods until either perfect performance is reached or the effectiveness of the features and actions available to the device is exhausted. At this saturation point the performance level of the device will plateau.This plateau effect is a consequence of the limitations of the fixed features and actions available to the adaptive device.

Evolutionary devices will show general improvement over their training period with varying rates of improvement. Their performance curves have the same general features as their adaptive device counterparts, except that they do not necessarily "plateau out" when the effectiveness of a given set of semantic relations is exhausted. New features and actions can be arrived at as long as the construction process is rich enough, and the device has enough time to sample enough alternatives.

Underneath the diagram are three time scales corresponding to the three feedback loops operating in the taxonomy: a measurement-computation-control loop, a testing-syntactic switching-performance loop, and a testing-construction-performance loop. Each has its own characteristic time constant. In general, any kind of adaptive loop must take longer than a single trial, but whether syntactic switching takes longer than construction is a function of the relative state of the two technologies. In present technologies and higher organisms, syntactic switching is much faster than construction of new semantic and syntactic primitives.

The performance characteristics of general evolutionary devices will depend upon which adaptive processes are fastest. If physical searches are slower than computational ones, the syntax will be adapting within a relatively stable, slowly changing semantic framework. If physical searches are faster, then the semantics will adapt to a relatively stable, slower changing syntax.

It should be noted that for simplicity we have assumed a stable measure of performance throughout--the task doesn't change. An adaptive or evolutionary device can only be as good as the performance measure employed to effect selection and thereby steer the device towards more optimal structures. Once the performance of the device exceeds the discriminatory capacity of the performance measuring process all adaptation ceases.

In cases where the measure of performance is changing, as in realms where there is competition between co-evolving actors, adaptability becomes even more valuable than in the static case, allowing the organism, device, or species to keep up with the demands of its changing environment (Conrad, 1983).

Arenas of adaptivity and adaptive possibility

The learning capacities of a given device type are limited to the parts of the device which are plastic and whose structure is contingent upon performance. The various capacities and limitations summarized in figure 11.2 reflect this.

device type	plasticity	capacities	limitations
formal-computational	fixed syntax	reliable execution of pre-specified rules	limited to pre-specified rules and states
formal-robotic	fixed syntax fixed semantics	reliable execution of fixed percept-action combinations	no feedback or learning from environment
adaptive	adaptive syntax fixed semantics	performance-dependent optimization of percept-action coordination	limited to percept & action categories fixed by the sensors & effectors
general evolutionary	adaptive syntax adaptive semantics	creation of new percept & action categories; performance-dependent optimization within these categories	time to construct & test new sensors & effectors may be very long

Figure 11.2 Relative capacities and limitations of the device types.

It should be obvious that formal devices will not of themselves realize either new syntactic or semantic relations. An adaptive device will not find new semantic relations, just as a syntactically-fixed evolutionary device will not find new syntactic relations.

The converse of this is also true. *Those parts of the device which are part of an adaptive loop can be used to find useful syntactic and semantic relations*. Since the class of all syntactic possibilities can be

easily circumscribed beforehand, an adaptive device will not find syntactic relations which are fundamentally new to us, but it will discover for us some relatively useful ones. This is the nature of the information we acquire by using an adaptive device.

Since the class of all semantic possibilities cannot be circumscribed beforehand, an evolutionary device *can* find unanticipated semantic relations. An evolutionary device can find some semantic relations that are relatively useful. Thus the information we acquire by using an evolutionary device may lead to unanticipated new functions. *Evolutionary devices are therefore functionally open-ended.*

This property of functional open-endedness will be discussed in more depth in the following two chapters.

Stages of problem-solving

The taxonomy of devices can be related to a general stage-theoretic account of problem-solving. In many cases, effective problem-solving can be seen in terms of the relaxation of constraints and the increase of possible solutions when creativity and innovation is required, and the tightening of constraints and the decrease of possible solutions when plain efficiency is demanded.

In general we start with a situation where the problem solution is wide open and proceed to narrow down the range of potential solutions through directed search.

Our first decision is always pragmatic in character. How do we know when we've achieved an adequate problem solution? What are our criteria? Without a clear idea of what we are after, we can go nowhere. This is always in some sense a purely subjective, *intentional* question, related to our needs and purposes, why we are dealing with the problem in the first place.

Second, we try to define the general terms of the problem, a semantic framework for interpreting the problem, the basic elements of our solution.

Third, if this seems to be adequate to the problem, we proceed to test out possible solutions within the semantic framework, by trying different parameters or combinations. We try to come up with an effective syntax for solving the problem, the right combination of elements. If we cannot find an adequate solution we may try other combinations or even go back to rethinking our general framework.

Fourth, if we do find a solution, we stop our search and write down the solution, thereby freezing out all the degrees of freedom involved in our semantic and syntactic searches. This will allow us to solve the problem more efficiently in the future, by avoiding a lot of searching through possibilities.

Problem-solving can be seen as the successive constraining of solution possibilities, from the selection of appropriate semantic relations, through evolutionary devices, to the selection of appropriate syntactic relations, through adaptive devices, to the efficient and reliable implementation of a solution, through formal devices. The procedure is find and freeze out appropriate semantics, find and freeze out appropriate syntax, implement the solution efficiently. This process is schematized in figure 11.3.

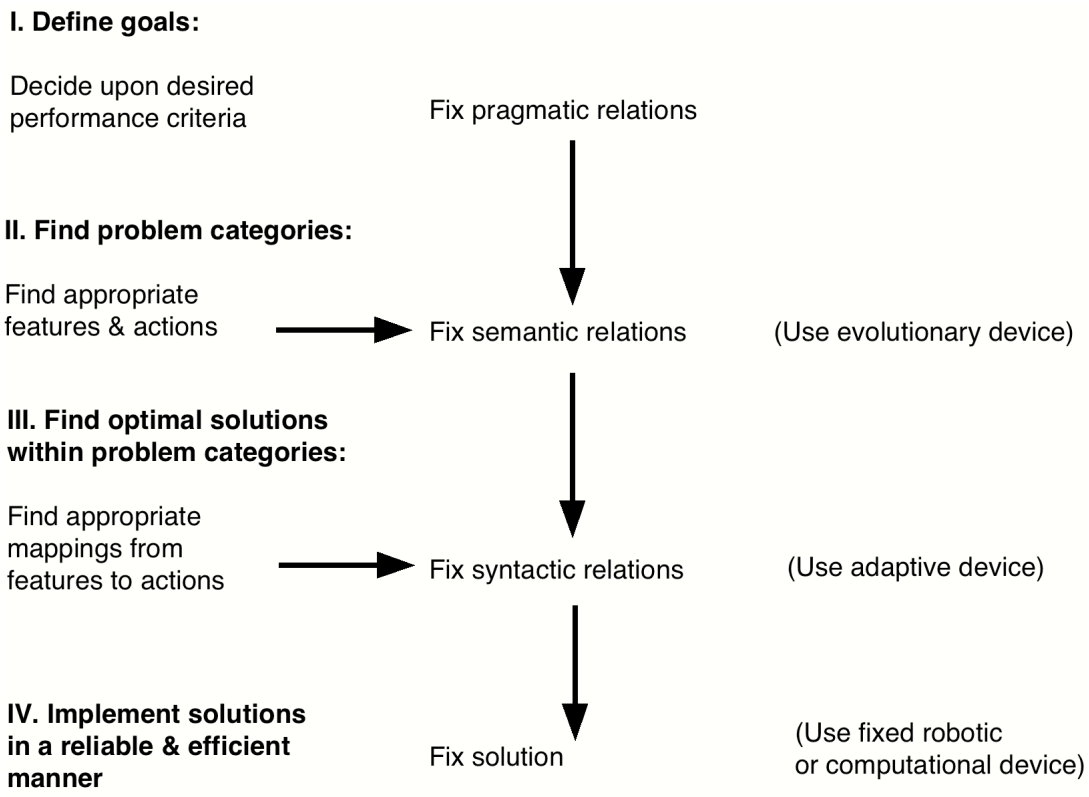

Figure 11.3 Types of adaptivity and stages of problem-solving.

Types of adaptive design strategies

Each device type also represents a strategy for design. Areas of plasticity and adaptability in a design represent pragmatic choices which have been deferred by the designer and given to the user.

Formal devices are embodiments of a designer-directed strategy of pre-specifying all behaviors. All order is imposed order. Most older, noninteractive computer programs and formal bureaucracies attempt to foresee all contingencies in advance and specify exactly what must be done in each instance. Those situations that are not foreseen are forbidden.

Adaptive devices embody a more flexible design philosophy in which the designer defines the overall framework of the design beforehand, leaving the choice of particular parameters to the user. In this way the user can, within limits, optimize the usefulness of the device based upon specific contexts. The design philosophy is one of limited or constrained choice for the user. Much interactive computer software nowadays is of this philosophy, allowing for the user to configure the software in a variety of ways.

An evolutionary design strategy allows the user choices in defining the overall framework for the design. The design philosophy is one of user participation from the earliest stages of planning so that the global aspects of the design are best adapted to the user's needs.

Because of the cost of searching for optimal constraints, no one design philosophy is best suited to all contexts. In contexts where most users have little or no interest in participation, the cost of testing and functional disruption is high, and high efficiency and reliability is necessary, such as the determination of postal routes or telephone switching patterns, outright total specification is warranted. Where users want some choices, but don't want to be overwhelmed by them, as in the case of computer software, a strategy of adaptive configuration of the machine to the user's needs is warranted. Where many users want to participate in *all* aspects of the system, as in urban planning or in participatory-

democratic institutions, all sorts of existing and imaginable possibilities must be left open, at least in the beginning of the design process.

Chapter 12. Emergence and open-endedness

> Anyone who looks at living organisms knows perfectly well that they can produce other organisms like themselves....
>
> Furthermore, it's equally evident that what goes on is actually one degree better than self-reproduction, for organisms appear to have gotten more elaborate in the course of time. Today's organisms are phylogenetically descended from others which were vastly simpler than they are, so much simpler, in fact, that it's inconceivable how any kind of description of the later, complex organism could have existed in the earlier one. (John von Neumann, 1949, pp. 481-482)

> The main point to realize is that all knowledge presents itself within a conceptual framework adapted to account for previous experience and that any such frame may prove too narrow to comprehend new experiences. (Neils Bohr, 1954, p.67)

The problem of emergence

The problem of emergence is useful in evaluating the open or closed nature of the devices in our taxonomy precisely because it relates to the problem of novelty in the world. If we want to enlarge our own capabilities and free ourselves of the burden of complete specification, our devices must be creative. If we want our devices to be creative in any meaningful sense of the word, they must be capable of emergent behavior, of implementing functions we have not specified. Our emergent devices must not be prisoners of our notational systems if they are to aid us in our own break-out.

Since all of our present biological and psychological structures and functions have emerged from simpler organisms, if we fully understood the process of biological emergence, we could replicate it in our devices:

> It would be very convenient if intelligence were an emergent behavior of randomly connected neurons in the same sense that snowflakes and whirlpools are emergent behaviors of water molecules. It might then be possible to build a thinking machine by simply hooking together a sufficiently large network of artificial neurons. The notion of emergence would suggest that such a network, once it reached some critical mass, would spontaneously begin to think. (Hillis, 1988, p. 175)

Traditionally, examples of emergence have come from four areas: emergent properties in physical state transitions, emergent functions in biological evolution, emergent ideas in the psychology of learning and emergent social structures, technologies and cultural innovations in social evolution.

> I think that science suggests to us (tentatively of course) a picture of a universe that is inventive [cp. Denbigh, 1975] or even *creative*; of a universe in which new things emerge, on new levels.
>
> There is, on the first level, the theory of emergence of heavy atomic nuclei in the centre of big stars, and, on a higher level, the evidence for the emergence somewhere in space of organic molecules.
>
> On the next level there is the emergence of life. Even if the origin of life should one day become reproducible in the laboratory, life creates something that is utterly new in the universe: the peculiar activity of organisms; especially the often purposeful actions of animals; and animal problem solving. All organisms are constant problem solvers; even though they are not conscious of most of the problems they are trying to solve.
>
> On the next level, the next great step is the emergence of conscious states. With the distinction of conscious and unconscious states, again something utterly new and of the greatest importance enters the universe. It is a new world: the world of conscious experience.
>
> On the next level, this is followed by the emergence of products of the human mind, such as the works of art; and the works of science; especially scientific theories. (Popper, 1987, p. 142)

All of these levels are related to the emergence of new scientific theories, to "evolving epistemologies" (Munévar, 1981; Campbell, 1987), to issues of physical and methodological reductionism and to the ontological status of scientific theories, issues which go to the heart of the philosophy of science. These

important issues are simply too complex to be adequately addressed here and will need to be dealt with elsewhere.

Physical emergence. A classic example of physical emergence is seen in the abrupt changes of phase that collections of water molecules undergo with temperature: between ice, water, and steam. Were we to have a simple model of the water molecules in their gaseous phase interacting as an ideal gas, we would not be able to predict that the gas will begin to change to the liquid phase slightly above 373 degrees Kelvin. Cooling another 100 degrees, we would not be able to predict from our model that ice crystals would form, much less the range of shapes that they would assume. These new phases and configurations would appear to us as emergent properties of water vapor.

A related example concerns the flow of traffic through a network of streets in a city. At low traffic densities the flow of cars changes relatively smoothly relative to the density until a critical density is exceeded and gridlock ensues. Traffic flow abruptly drops to nearly a complete standstill. One could say that gridlock is an emergent behavior of automobiles and their drivers at higher densities.

Biological emergence. Perhaps the most striking aspect of biological evolution is the overall increase in the morphological complexity of living things over time. If we look at the number of different cell types in organisms, we can see the successive appearance of organisms with more and more specialized cell types (Bonner, 1988). Enormously complex structures and functions have evolved from simpler organisms. The problem of emergence originally referred to the evolution of new biological structures and functions, which had no obvious explanation in terms of existing ones (Morgan, 1931).

Psychological emergence. Subjectively, we often experience revelations or new ideas which seem to come from nowhere. The process of recognizing a visual pattern which at first seems abstract and disconnected but suddenly gels into a coherent Gestalt, is a perceptual analogue of the physical change of phase. A classic, often-cited example is the account of Poincaré, who experienced sudden, unexpected flashes of insight while trying to solve mathematical problems:

> Most striking at first is this appearance of sudden illumination, a manifest sign of long, unconscious prior work. The role of this unconscious work in mathematical inventions seems to me to be incontestable, and traces of it would be found in other cases where it is less evident. Often when one works at hard questions, nothing good is accomplished at the first attack. Then one takes a rest, longer or shorter , and sits down anew to the work. During the first half hour, as before, nothing is found, and then all of a sudden the decisive idea presents itself. (Poincaré, 1913, p.389)

In the psychological case, the conscious subject suddenly creates a new idea, one which was not directly derived from pre-existing ones.

Contemporary theories of emergence

Emergence deals with the origins of complex order in the world. There are two major ways to view the origin of order: order-from-order accounts vs. order-from-chaos accounts. Chaos here has its original meaning as formlessness or structurelessness. These accounts correspond to the images of "the crystal vs. the flame," (Piatelli-Palmarini, 1980), to completely different epistemologies of order and its origins (Maruyama, 1977).

On the contemporary scene, there are three major conceptions of emergence, *computational emergence*, *thermodynamic emergence*, and *emergence-relative-to-a-model*. Some of the major features of these perspectives are presented in figure 12.1.

Theory	Account of the origins of order	Ontology	Research program
Computational Emergence Mathematically based	**Order-from-Order** Macro-order from micro-determinism Macro-indeterminism through mathematical chaos	Discrete Universe Monism Microscopic Deterministic Rules	Realize emergent behaviors through cellular automata and evolutionary computer simulations (Langton, Toffoli)
Thermodynamic Emergence Physically based	**Order-from-Noise** Discrete macro-structures (symbols) from continuous micro-processes New structures emerge through fluctuations	Continuous Universe Monism Continuous Physical Laws	Develop a thermodynamic theory to describe how emergent structures can arise far from equilibrium. Apply this to biological & social systems & upwards (Prigogine, Iberall)
Emergence relative to a model Biologically based	**Form-from-formlessness (Order-from-Chaos)** Processes of linking symbols to the world result in new functions and extend the realm of symbolic activity	Symbol-Matter Dualism Dualism between Definite (measured) & Indefinite (un-measured)Entities	Realize emergent functions through the construction of semantically adaptive devices, augment the capabilities of the observer (Pattee, Rosen, Pask)

Figure 12.1. Three major perspectives on the problem of emergence.

Alternatively, the three conceptions can be seen as mathematically-based, physically-based, and biologically-based notions of emergence. Mathematically-based emergence concerns itself with the formation of *new formal structures,* strings of symbols and their behaviors.[1] Physically-based emergence concerns itself with describing the origins of *new physical structures*, like Benard cells and auto-catalytic chemical networks. Biologically-based emergence concerns itself with the origins of *new functions,* like the ability to sense light, to breathe on land, or to fly. The first two are structurally-based conceptions, while the last is functionally-based.

Computationalist conceptions of emergence

The computationalist conception of emergence is embraced by many connectionists and most of those engaged in artificial life simulations. Rather than arising from without, macroscopic order emerges from an already existing microscopic order.

> The 'key' concept in Artificial Life is *emergent behavior*. Natural life emerges out of the organized interactions of a great number of nonliving molecules, with no global controller responsible for the behavior of every part. Rather, every part is a behav*or* itself, and life is the behav*ior* that emerges out of all the local interactions among individual behav*ors*. It is this bottom-up, distributed local determination of behavior that Artificial Life employs in its primary methodological approach to the generation of lifelike behaviors. (Langton, 1989, pp.2-3)

The basic assumptions of this position are strongly realist, mechanist, and reductionist: that all behavior which is apparently is *really* reducible to the logical consequences of unobserved, rule-like micro-behaviors.

[1]In a sense computationalist emergence and the catastrophe theory Rene Thom (1975) have this in common: they are primarily about ideal, platonic structures and secondarily about the description of nature. Where they differ is that computational emergence embraces a discrete picture of the universe, and therefore discrete mathematics, whereas catastrophe theory embraces a continuous image, and therefore utilizes continuous mathematics.

> The rules that govern the forces between water molecules seem much simpler than crystals or whirlpools or boiling points, yet all of these complex phenomena are somehow consequences of those rules. Such phenomena are called *emergent behaviors* or the system. (Hillis, 1988)

An implicit assumption about these "rules which govern water molecules" is that they are rule-like, deterministic. It is an article of faith within the computationalist worldview, that all physical processes can be reduced to computation-like, primitive operations at some rock-bottom, exhaustive level of description. This belief places the computationalist worldview very close to Laplacian determinism, with discrete microscopic rules replacing the continuous deterministic laws of classical physics. Within this worldview, mathematical chaos explains how apparent macroscopic disorder can arise from microscopic order, eliminating the apparent incongruity between a rule-governed, computation-like micro-order and the noisy stochasticity of everyday life in the macroscopic realm. It obviates any need to resort to nondeterministic or indefinite microscopic processes. However, it is an open question whether *any* microscopic determinism can be reconciled with macroscopic emergence in any reasonable sense of the word (Denbigh, 1975; Prigogine, 1980; Klee, 1984; Rosen, 1985b). Denbigh (1975) has argued that any completely deterministic view of the world is fundamentally incompatible with that of an emerging order.

> Let us speak about *inventive processes* as being those which result in the production of something which is novel. An essential condition for inventiveness in this sense is the *absence of necessity*. One cannot speak of genuinely new things coming into existence unless...they are both different from and not necessitated by anything existing beforehand. Thus if there are processes fully determined in advance, these could not be regarded as inventive. However ...there are certainly many processes which whose outcome is not predictable even in principle, and where it would, therefore be gratuitous to suppose that they are 'determined'. Such processes would, therefore, qualify as being inventive if they also result in the production of novelty. (Denbigh, 1975, p. 153; *Notice that a continuous production of *the same* entities, e.g. hydrogen atoms, would not be classed as inventive except at its first occurrence.)

Denbigh's definition notwithstanding, if one adopts computationalist ontology, computer simulations are isomorphic in their processes to those of the physical universe itself at its rock-bottom level, where the deterministic microscopic rules operate. Since physical systems have emergent processes and computer simulations are isomorphic to physical systems, then on some level, maybe many levels well above the simulation's rock-bottom level, computer simulations will also exhibit emergence.

> There has been a recent renewal of interest in emergent behavior in the form of simulated neural networks and connectionist models, spin glasses and cellular automata, and evolutionary models. Each of these is a model of some real system. For neural networks and connectionist models, the system being modelled is a collection of biological neurons, such as the brain; for spin glasses it is molecular crystals. Cellular automata and evolutionary models are based on the ontogenesis and phylogenesis of living organisms. In all these cases, both the model and the system being modelled produce dramatic examples of emergent behavior. (Hillis, 1988, p.176)

It was argued earlier that formal-computational devices cannot create new primitives. *Any process which can be completely simulated via computations will not generate new primitives.* Computer simulations of any sort, be they simulations of neural networks, connectionist networks, spin glasses, cellular automata, or evolutionary models, will not create properties which were not encoded in the simulation from the start. There was some *very* limited debate over this point at the First Artificial Life workshop, but no plausible strategies for generating new primitives in computer simulations was suggested. Appendix 5 discusses specific strategies that have been proposed and their shortcomings.

It cannot be disputed that *some* physical systems, when viewed through the appropriate observational frame, will appear this way, and a primary aim of science is to expand the range of phenomena which can be described by means of deterministic rules. However, it is gratuitous to claim that the universe is an enormous computational device without at least giving some account of what the primitives might be

and how they might be observed. And the experience of quantum mechanics makes us suspect that at least *some* processes cannot be so reduced.

The general strategy pursued by the computationalists is to design a simulation with enough primitives and rules of interaction such that the behavior of the simulation is not easily foreseen by an external observer. These simulations can be quite visually compelling and thought-provoking, especially if viewed in real time (Tamayo & Hartman, 1989). The observer then notices that higher level patterns have formed: cycles, stable configurations, stable operations, boundaries, phase transitions, attractors, etc (e.g. Kauffman, 1984). These higher level patterns are then literally said to have emerged over the course of the simulation, and *the emergent behavior is consequently attributed to the device itself.*

> The molecular logic of life is a dynamic distributed logic. An initial set of operators and operands goes to work producing more operators and operands, which immediately enter the ongoing logical 'fray'. Some of these new operators and operands are distributed as new initial sets in the process of self-reproduction. This dynamical character of the molecular logic of life is unlike a typical formal logic which, although it provides an initial set of operators and primitive operands, has no internal dynamics of its own. Instead, formal logics of the standard variety provide people with convenient tools, but they are passive tools not active ones. They must be applied by something or somebody outside the system. (Langton, 1986)

What is left out here is that the higher level patterns are patterns which must be recognized by the human observer. No new rules come into play which were not in some sense prespecified. No behavior arises which is not a logical consequence of the simulation rules and the initial state. There is no doubt that all of these simulations spark human creativity and catalyze new ideas, which makes them quite worthwhile activities. They are indispensable for the understanding and design of complex systems. We should continue to pursue them for these reasons. But it is a mistake to believe this to be a property of the device.

Henri Poincaré recognized that insight resides in the minds of mathematicians, not in the symbol manipulations of their instruments of calculation:

> It is for the same reason [the aesthetic satisfaction of mathematical elegance] that, when a somewhat lengthy calculation has conducted us to some simple and striking result, we are not satisfied until we have shown that we might have foreseen, if not the whole result, at least its most characteristic features. Why is this? What is it that prevents our being contented with a calculation which has taught us apparently all that we wished to know? The reason is that, in analogous cases, the lengthy calculation might not be able to be used again, while this is not true of reasoning, often semi-intuitive, which might have enabled us to foresee the result. This reasoning being short, we can see all the parts at a single glance, so that we perceive immediately what must be changed to adapt it to all problems of a similar nature that may be presented. ...
>
> What I have just said is sufficient to show how vain it should be to attempt to replace the mathematician's free initiative by a mechanical process of any kind. In order to obtain a result having any real value, it is not enough to grind out the calculations, or to have a machine for putting things in order: it is not order, but unexpected order, that has a value. A machine can take hold of the bare fact, but the soul of the fact will always escape it. (Poincaré, 1913, in "The future of mathematics," Chapter II of Science and Method, pp. 369-382)

Poincaré upheld the usefulness of playing with symbol manipulations (in our case simulations) while recognizing that the machine by itself cannot create new mathematics because it is incapable of creating new conceptual primitives. Recall also that the creation of ever more powerful logics in history of mathematics cannot itself be a completely mechanical process (Fodor, 1980; Piaget, 1980).

Another difficulty with the computationalist definition of emergence is its absolute ubiquity. If emergence is to be defined as the result of a computation for which we cannot readily foresee the answer, then virtually *all* computations become emergent processes. In what sense does the act of dividing 1792 by 7 and arriving at 256 exemplify emergence? If the computationalist view is taken to its logical conclusion, *emergence* becomes synonymous with *change,* and its meaning becomes debased.

Finally, while Langton makes the distinction between dynamic, distributed logics and static, sequential ones, *criticizing the latter as being merely passive tools*, the same criticisms can be made of distributed logics. Cellular automata must always start with an initial set of operators and primitive operands, and any other operators or operands which come about must be reducible to combinations on this initial set. The newly formed operators and operands are therefore not *primitive* :

> This concept of emergence, which we might call nomic-emergence, can be more precisely characterized by saying that in certain systems the micro-properties or micro-constituents are brought into a novel relational structure in virtue of their integration on the higher level organization. This new relational structure is responsible for new law-like regularities to the behavior of the system at that higher level of organization. This new relational structure is novel relative to the kinds of relational structures found at lower-levels and, in turn, responsible for a broadened range of regularities in behavior the system has been capable of exhibiting. The new regularities are in this sense grounded in the new relational structure formed by the higher-level integration of the micro-properties.
>
> But in what sense are these new regularities emergent? To be sure they may be regularities and structures of a type not found on lower levels of organization, but it seemed to some (Nagel, 1961, pp.367-74) that this fact by itself would not justify the label 'emergent' if they had been predictable on the basis of a thorough understanding of those lower-levels of organization." (Klee, 1984, p.46; Nagel, E., (1961), *The Structure of Science*. New York: Harcourt, Brace & Court)

More important than semantic quibbling, the assumption of pre-existing, rock-bottom primitives many levels down denies the possibility of the emergence of new primitives entirely. As has been argued, computer simulations do not create new primitives by themselves, and if they are images of the universe as a whole, then the universe cannot create new primitives. Thus, if the concept of emergence is to be salvaged and made consistent with the rest of the computationalist account, then the basic concept of emergence itself must be altered. An example of this semantic shift can be seen if we examine closely the kinds of processes which are held to be emergent by computationalists. This is the same shift that was made by some information theorists (e.g. McKay, 1969) and early cyberneticians towards explaining terms like "meaning" and "creativity" in purely formal terms:

> McKay, like other systems theorists, appears to be laboring under a philosophical confusion that is obscured by his scientific terminology. If by *originality* or *creativity* we mean the bringing into being of something that did not exist before, it appears to me that he transfers the problem from one level to another without solving it; that is, he appears to assume that original information consists of the combination of preexistent elements in ways that have not occurred before. But if the elements are finite and determinate, so is the number of their combinations and permutations, though the number may be very large. Creativity [in McKay's terms] is defined as the ability to bring about specific permutations of preexistent elements that have not been noticed before. But if the combinations or permutations have not been previously noticed, this does not mean that they have not previously existed, at least in principle. But then originality is defined as the combination of preexistent elements combined with the ability to notice that a specific combination is unusually fruitful or valid. The meaning of originality or creativity has been shifted away from its initial sense of bring about of something that did not exist before. (Lilienfeld, 1978, pp.87-88)

Reconstructed in this way to exclude *de novo* primitive creation, the computationalist concept of emergence loses much of its usefulness, forcing us to conclude, with Hillis, that "The concept of emergence in itself offers neither guidance on how to construct such a[n intelligent] system nor insight into why it would work." (Hillis, 1988, p.176) Like Hillis, we must conclude that there is no particular reason to believe that a large number of computational elements, randomly connected will begin to exhibit intelligent behavior.

What would have to happen for there to be such *de novo* emergence of new structures in a cellular automaton simulation? We would expect physically coherent, rate-dependent dynamics of interaction to could come into play during the course of the simulation, ones which were not specified by the designer at the simulation's start. These would have to be rules or dynamics which could change without invoking yet another prespecified rule. An example of this would be if the global behavior of a cellular

automaton began to modify or constrain the local rules of interaction, without the global-local interactions being specified by the simulation's designer from the start. In the Game of Life, the local rules of a node might spontaneously change based upon whether it was in an area completely enclosed by "on" nodes. New patterns would arise. The simulation would then be open-ended since new stable symbol primitives, and hierarchies of interaction could arise which were not preprogrammed (Pattee, 1973; Rosen, 1973). The behavior of the system would no longer be reducible to the prespecified local rules of interaction. This is not very far from the *intuitive* conception of emergence held by the computationalists (how easy things would be, if only it were true!). These strange simulations would be genuinely "nonlinear," in Langton's (1989) sense of the term. But if such interactions are allowed in a device, as they must be to allow for the formation of new primitives, the system is no longer a formal system carrying out computations in the sense defined in chapter 7. Suddenly the state transition rules would be ambiguous and the input-output function would be rendered indeterminate. Results would not be replicable, and the central, defining characteristic of formality would be lost. We would be outside the computational realm entirely. *Our computer would no longer be functioning as a computer, and our simulation would no longer be formally reproducible*. We can have emergent devices if we give up the deterministic, symbolic nature of the devices, and we can have well-behaved deterministic computer simulations as long as we give up the hope of making them emergent, but *we cannot have both at the same time*.

Thermodynamic conceptions of emergence

The thermodynamic conception of emergence sees emergence as the formation of new physical structures in the world at large. Theories of thermodynamic emergence generally attempt to describe emergent processes in terms of differential equations. Unlike the strong computationalist conception of emergence, however, thermodynamic theories do not claim to actually realize emergent structures in the computation of the equations describing the physical system. Because this dissertation is mainly concerned with the *generation of emergent functions* rather than theoretical explanations of emergent structures, theories of thermodynamic emergence are less directly relevant to our present purposes than the claims of the computationalists. And because strong claims are not made regarding the actual generation of emergent structures, many of the criticisms made of computational emergence need not be made for the thermodynamic approach.

The thermodynamic approach includes, among many others, the nonequilibrium thermodynamics of Prigogine and co-workers (1980, 1984), the thermodynamic fields of Iberall (1977), the hypercycles of Eigen and Schuster (1978, 1979), Denbigh's "inventive processes" (1976), the dynamical morphologies of Abraham and Shaw (1976, 1984), and the dynamically-based network analysis of Rosen (Rosen, 1970, 1971, 1981; Minch, 1988). Some of these theories arise out of classical mechanics (emphasizing trajectories), some out of classical thermodynamics (emphasizing probability distributions). Generally speaking, the analyses deal with ensembles of micro-states, the relation of microstates to observable macro-states, the relation of energy flows to the probability distributions of ensembles of both micro- and macro-states, the formation of dissipative structures, and the "dynamical morphology" of attractors, separatrices, limit cycles, and various other species of periodic and chaotic trajectories. The primary phenomena to be explained by thermodynamic emergence is pre-biotic evolution, where complex molecular reaction cycles and networks arise out of the chemical flux to form complex, self-propagating structures. Out of the analysis of chemical systems come concepts which are used to explain biological and social structures.

Because the approach is so broad and the concepts so general, there is a good deal of thermodynamic analysis pertaining to the other two conceptions of emergence. Thermodynamic emergence and computational emergence overlap where the formal structure of the theory is emphasized and/or where discrete, deterministic micro-processes can be postulated (e.g. Toffoli, 1982, 1984). On the

mathematical end, theories of chaos, bifurcations, and dynamical morphologies are often simulated using cellular automata and other discrete mathematical instruments (see papers in Holden, ed., 1986; Tamayo & Hartman, 1989; Toffoli & Margulis, 1987). Thermodynamic emergence and emergence-relative-to-a-model overlap where considerations involve the dissipative nature of symbolic function (e.g. Pattee, 1973b, 1974) the thermodynamic origins of symbols (e.g. Pattee, 1969, 1973a, 1973c), reversibility and irreversibility in modelling relations (e.g., Rosen, 1985a, 1985b), the dynamical analysis of nervous system coordination (Kugler *et al*, 1984) and the relationship between thermodynamics and evolution (e.g. Yates, 1977; Wicken, 1988; Collier, 1988; Depew & Weber, 1988; Swenson, 1989).

As was noted in chapter 3, the various thermodynamic theories still have not been effectively unified under a set of commonly understood concepts and assumptions. When a comprehensive theory of emergent thermodynamic structures is finally achieved, we shall presumably have a very powerful theory which can potentially explain the origins of symbols and their associated functions. Eventually, when this happens, we will want to consider two general related problems.

The first is *the problem of emergent function,* or, in other words, given a theory of emergent structures, how does one use these emergent structures to acquire emergent function? Once such a theory had been developed, we could use it to design the construction systems of evolutionary and general evolutionary devices. In this way a theory of emergent structures could be integrated with the theory of emergent functions developed in this dissertation.

The second problem involves the *empirical recognition of emergence* in the physical world. Given a thermodynamical theory of how new structures arise, how would we recognize one if we saw one? To do this we must have some means of linking the terms of our thermodynamical theories to real world observables, or else our theories will be pragmatically useless, incapable of making useful predictions.

Both of these problems are addressed by the observer-centered conception of emergence, one which sees emergence as behavior relative to a model. Ultimately we want a theory which will guide us in our construction of fundamentally new ways of observing and acting on the world.

The observer-centered conception of emergence

To contemplate such a theory, however, we must recognize the functional value of creating new measurement and control linkages with the world. We must be able to see how it might be possible for new observables and new actions to come about *from the inside out and appreciate what this means.*[2] In

[2]What stands in the way of our seeing this process and fully appreciating its importance is a pervasive realist attitude which renders almost any connection with observables or observers as unnecessary. In the philosophy of science we have the parallel of the syntactization of meaning going on in the formalization of observation (e.g. as in Carnap). This general attitude, that linkage to an external world is unnecessary, is particularly prevalent in the computationalist literature. In some places the platonic desire to free theory from its empirical fetters reaches the point where *all* observables are deemed unnecessary. Arguments are made that computers by themselves can in effect do science without the necessity of empirical observation. In these arguments one can find a rather inexplicably narrow-minded restriction of what counts as an "observable" to "human biological observables," i.e. those properties apprehended by means of the five senses normally available to the human organism (Churchland, 1985, pp. 42-43; Thagard, 1988, pp.150-151; Laing, 1989, pp.56-57). It is then argued that, since all of the human sense organs could be ablated and artificial sensors substituted in their place, "observables" are no longer necessary for connection with the world, since connections can be made by other means than natural biological sense organs. Correspondingly, it is held that for computers outfitted with these artificial sensors and effectors "the *observable* world is an empty set" (Churchland, 1985, p.43). This kind of theory completely precludes the emergence of new observables by obscuring the meaning of the the term "observable."

this regard the experience of Helen Keller offers a dramatic example. By her own account, the day she began to connect the signals with her other senses marked a striking expansion of her consciousness:

> The biography of Hellen Keller, the blind and deaf mute whose early development was frustrated almost to the point of neurotic breakdown, throws a partial light on the origin of language. Though it has been often cited, it remains too important to be passed over. For almost seven years she lived in darkness and mental isolation, not merely without clues for identifying the world about her, but often full of savage rage because [she was] unable to articulate or communicate her own feelings. Intelligible messages between her and the outside world neither came nor went. ...
>
> Then for Helen came the famous moment when she was suddenly able to couple the sensation of water with the symbolic taps made by her teacher on the palm of her hand. With that, the meaning of a word dawned on her: she found a way of coupling symbols with things, sensations, actions, events. The over-used term 'break-through' surely applies to that moment." (Mumford, 1966, p.82)

Helen Keller's breakthrough is paradigmatic for observer-centered emergence. Her connection of the "symbolic taps" to the sensation of water on her hand gave her a means of accessing what the teacher was perceiving, enabling her to begin to experience the world through others' bodies. New sensory modalities were suddenly available to her in a way not unlike what happens in the evolution of a new sense organ.

The realists assume a vantage point of ontological omniscience, and the particular problems of the limited nature of the observer disappear.[3] But these are *our* problems! In a sense, we are all like Helen Keller, seeking to expand our worlds. The observationalist account assumes the vantage point of the observer, which pragmatically is our own epistemic situation. Such an observer-centered perspective illuminates the problems we as limited observers face in trying to cope with the world. Of much more value to us in our attempts to design devices with emergent properties is "How do we design a device which will have behaviors which are fundamentally novel with respect to us, thereby enlarging *our* world?" If this is to occur, we must be able to design devices which violate our expectations, thereby exposing us to aspects of the world hitherto unknown by us. This concept of subjective novelty, however, remains a vague concept until we are able to specify an observational frame through which we can unambiguously determine whether a given behavior has corresponded to our model of it.

> I have tried, however, to indicate a general attitude suggested by the serious lesson we have in our day received in this field and which to me appears of importance for the problem of the unity of knowledge. This attitude may be summarized by the endeavor to achieve a harmonious comprehension of ever wider aspects of our situation, recognizing that no experience is definable without a logical frame and that any apparent disharmony can be removed only by an appropriate widening of the conceptual framework. (Bohr, 1954, p. 82)

Once we have constructed such a framework, in the form of a model and its associated observational frame, we can precisely define and communicate our subjective expectations within the frame of the model. Once this has been accomplished, we have the means of unambiguously recognizing when the expectations of the model have been violated, when an emergent event has occurred.

Emergence relative to a model

Robert Rosen has proposed the definition of emergence as *the deviation of the behavior of a natural system from a model of it*.

---

[3]Arguments related to those of note (2) above have led some (even Cassirer) to contend that the senses are irrelevant because Helen Keller was able to reach extraordinary levels of cognitive development without them (Cassirer, 1944, pp. 36-39). But these accounts neglect the substitute sense organs of Ms. Keller's human companions and the facility of communication by touch, which allowed Ms. Keller to make their senses her own.

> From our point of view, as we have stated above, it will appear that *new properties of the system have emerged;* indeed, the phenomena of emergence so prominent in biology and elsewhere are nothing but the bifurcations between the behavior of a complex system and a simple model of it. (Rosen, 1985a, p.337)

For Rosen emergence results from the incompleteness of our models. *Models are impoverished descriptions of the world.*

> ...A natural system is essentially a bundle of linked qualities, or observables, coded or named by the specific percepts which they generate, and by the relations which the mind creates to organize them. As such, a natural system is always incompletely known; we continually learn about such a system, for instance by watching its effect on other systems with which it interacts, and attempting to include the observables rendered perceptible thereby into the scheme of linkages established previously. A formal system, on the other hand, is entirely a creation of the mind, possessing no properties beyond those which enter into its definition and its implications. We thus do not "learn" about a formal system, beyond establishing the consequences of our definitions through the application of conventional rules of inference, and sometimes by modifying or enlarging the initial definitions in particular ways. (Rosen, 1985)

We can only have a finite number of observables, while the world at large has an indefinite number of properties:

> At this point we must be clear about how a 'system' is to be defined. Our first impulse is to point at the pendulum and to say 'the system is the thing down there.' This method has a fundamental disadvantage: *every material object contains no less than an infinity of variables and therefore of possible systems*. The real pendulum, for instance, has not only length and position; it also has mass, temperature, electric conductivity, crystalline structure, chemical impurities, some radio-activity, velocity, reflecting power, tensile strength, a surface film of moisture, bacterial contamination, an optical absorption, elasticity, shape, specific gravity, and so on. Any suggestion that we should study "all" the facts is unrealistic and actually the attempt is never made. (Ashby, 1954, pp.39-40)

Emergence relative to a model, then is a result of the finite and hence incomplete character of all models of the world. At some point in time we can, if we are fortunate, construct a model which will deterministically capture the behavior of the physical system. The behavior predicted by the model will, for some period of time, correspond to the observed behavior of the physical system, because it was constructed to do so. But eventually, if one waits long enough, all physical systems will diverge from their models, but some will diverge before others. Physical systems can thus be sorted out according to whether they will exhibit emergence over some finite observational period. Those which do not will appear to conform to their models and will thus be classified as formal devices. Those which do are *emergent devices.*

There are always interactions going on in the world which are unrepresented in the model, which are sources for the divergence of the natural system from its model. The divergence is inevitable because of the inherent time-independence and structural stasis of a formal model versus the inherent time-dependent structurally dynamic nature of matter. In this context, the Bergsonian distinction between a motion-filled pulsating, living world and our unchanging, formal encodings of that world becomes relevant:

> *Time is invention or it is nothing at all.* But of time-invention physics can take no account, restricted as it is to the cinematographical method. It is limited to counting simultaneities between the events that make up this time and and the positions of the mobile T on its trajectory. It detaches these events from the whole, which at every moment puts on a new form and which communicates to them something of its novelty. It considers them in the abstract, such as they would be outside of the living whole, that is to say, in a time unrolled in space. It retains only the events or systems of events that can be thus isolated without being made to undergo too profound a deformation, because only these lend themselves to the application of its method. Our physics dates from the day when it was known how to isolate such systems. (Bergson, 1911, p. 371)

Emergence and the taxonomy of adaptivity

The taxonomy of devices presented in chapters 6 through 11 can be related to Rosen's concept of emergence. When the behavior of the physical system, in this case the device itself, bifurcates from the behavior of the model, another model will have to be constructed which will capture subsequent behavior of the physical system/device. The general ways that models can change are two: by means of changing the state-transition rules of the computational part of the model or by changing the observables via changing the measuring devices.

> The characteristic feature of emergent novelty is the need to pass to a new mode of system description after the emergence has occurred [to be able to fully capture the phenomena again]. Such a new mode of description is typically characterized by quite different observables than those appearing in the prior description and/or by new linkage relationships between previously defined observables. In all dynamical theories, there is simply no visible source for such new observables; this has always been the basic difficulty. (Rosen, 1978, p.91)

In Rosen's terms, "linkage relationships between previously defined observables" correspond to the syntactic rules of the computational part of the model. We will call the situation where changing the computational part of the model is sufficient to recapture the behavior of the physical device *syntactic-emergence*. Only new syntactic linkages need be formed. We will call the situation where adding new observables is necessary to recapture the behavior of the device *semantic-emergence*. These types of emergence correspond to device types in the adaptivity taxonomy. *Formal devices are nonemergent. Adaptive devices have syntactic-emergent behavior. Evolutionary devices have semantic-emergent behavior.*

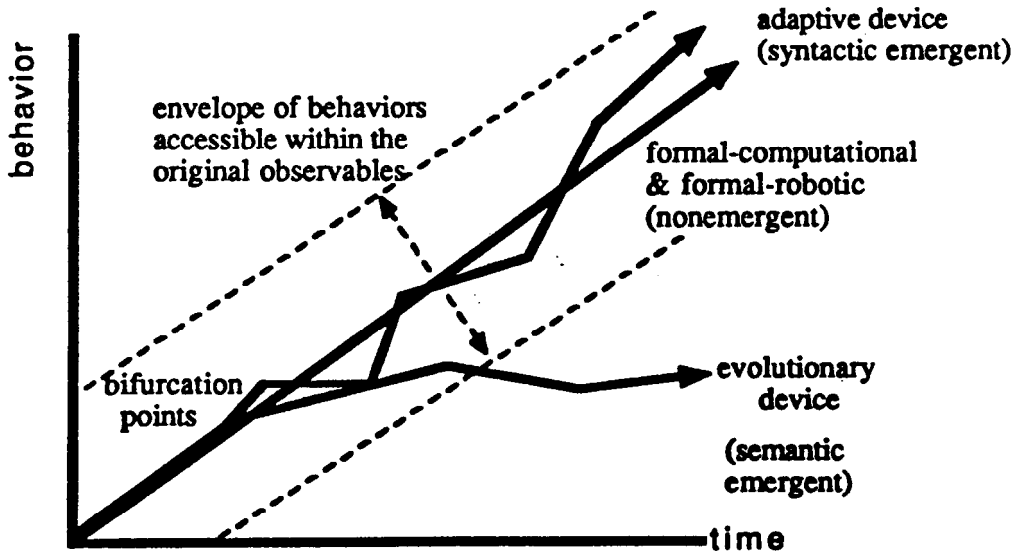

Figure 12.2 Divergence of the behaviors of adaptive devices from fixed models of them.

In order to model a formal-computational device, no change in the model is needed since by definition the device will conform to its specified state transition rules, and consequently there will be no bifurcation of device behavior from the model behavior.

In order to model an adaptive device undergoing training and performance improvement, one would not need to change the observables of the model; one would need only to change the state-transition rules of the old model.

In order to model an evolutionary device over the course of training, changing the state-transition rules of the old model would not be sufficient to capture the behavior of the device. Adding new observables to the model would be necessary for the new model to do this.

Property	Device type	Model change needed
Nonemergence	**Formal-computational** **Formal-robotic**	Device reliably conforms to model's expectations; no changes in model parameters are needed.
Syntactically emergent	**Adaptive**	Device diverges from model; but only adjustment of model parameters is necessary to recover full predictability.
Semantically emergent	**Evolutionary** **General evolutionary**	Device diverges from model; parameter adjustments alone are inadequate; new observables are needed to predict subsequent behavior.

Figure 12.3 Relationship between types of adaptivity and types of emergence.

These correspondences are presented in figure 12.3. A model, M with its associated observable set, O , and its set of state transition rules, ST, which make up its formal part, describes a given type of device. If a bifurcation occurs between the behavior of the device and the model's predicted behavior, two kinds of responses are possible to recover the predictability of the model, by changing the formal part (change ST) or change the observables (change O).

Emergence and the amplification of the observer

The new observable or control found by an evolutionary device can be used by the observer as a means of expanding the measurement and controls available to him/her, just as Helen Keller was able to use the senses of other people around her via a touch code. In Rosen's terms, she succeeded in linking her one touch observable to many others, switching the linkage according to what sense organ others told her they were using in touch code.

> From this point of view, the establishment of a linkage between a pair of previously unlinked observables is a prototypic example of an emergent phenomenon. The establishment of a linkage makes available an *alternate description*: a state previously describable by observables defined on its initial set of states becomes, after a linkage is established, describable by a new observable which has become linked to the original one. A subsequent dynamical interaction involving either of the newly linked observables can be described either description as long as the linkage persists; when the linkage is broken again by subsequent dynamics, we may pass to the new description. (Rosen, 1978, p.92)

If Ms. Keller was communicating with someone describing what they were seeing, her available descriptions would be either the touch code description or an alternate one in the terms of the other person's description. Once the communication link was broken, however, she would lose access to the other person's senses, but she would still have knowledge of the expanded set of perceptual distinction available to her when she re-established communication. In this way her model of the world was able to constantly expand. We undergo a similar, if less dramatic, process when we meet new people, talk with

them, and begin to see the world through their eyes. New distinctions are made available to us via new linkages.

If we have access to the internal states of our evolutionary device, we can similarly establish a linkage with the device and its new observables, thereby increasing the number of observables we have at our disposal. As was discussed in chapter 10, when we do this, our syntactic boundaries expand outward and the loci at which the semantic relations are defined, where the measurements are made and the controls performed, become the device's sensors and effectors. As Pask noted, we encounter a boundary definition problem when we attempt to observe this process from without:

> Let us distinguish between that sort of machine that is made out of known bits and pieces, such as a computer (in terms of which we can make models of most physical assemblages, those normally studied in physics, and perhaps, some of those we normally study in biology), and a machine which consists of a possibly infinite number of components such that the function of these components is not specified beforehand. In other words, these "components" are simply "building material" which can be assembled in a variety of ways to make different entities. In particular the designer need not specify the set of possible entities.
>
> Now, to say that this sort of machine will make sense organs is illustrative, because a sense organ is a rather special component construction in the sense that it specifies the boundary between the machine and its environment, and if the machine constructs its own sense organs out of the building material this boundary is apt to change continually. Further, it helps us to understand why we find it difficult to observe these self-organizing systems, namely, because, looking at them in a single reference frame, in the capacity of scientific observers we cannot lay down definitions which allow us to compartmentalize the functional components of the machine or even the machine itself.
>
> To see that this illustrative model is non-trivial you must recall that we reward the system without specifying, for example, that it will be rewarded for assembling component A and component B. We simply say it will get more reward (meaning it will be allowed to use more components) if a structure that acts like a sense organ is constructed. Clearly, unless you are prepared to take into account a variety of different ways of describing this machine (in different reference frames) you can't make sense of it, and this is what I meant by saying we must look at a self-organizing system as many coexisting models, mechanical, electrical and so on--rather than a single model--such as we have in a computer simulation. (Pask, 1959, p.262)

Evolutionary devices, those with adaptive semantic parts, are thus qualitatively different from all other devices in that their functional boundaries are not well-defined. This is a consequence of the partially defined nature of measurement and control processes, as opposed to the completely defined nature of computations.

Definiteness, indefiniteness, and functional replicability

We can repeat a computation exactly, but we can never repeat a measurement or a control exactly. The function of computation, being a completely symbolic one, can be completely described in terms of observed state transitions and, due to its definite symbolic nature, this behavior can be replicated *exactly*. No matter what the underlying mechanism, if the observed state transitions are the same, the same computation is being performed.

A measurement, however, being an incompletely defined semantic relation cannot be so exhaustively characterized by symbolic rules. This is why measurements are irreducibly distinct from computations within a particular observational frame. Probabilities can be assessed for the different possible state transitions of the measurement, but these are not sufficient to exactly replicate the observed behavior of the measuring device. Different measuring devices implementing different measurements can nevertheless exhibit the same state-transition probabilities. While the results of measurements can be compared to those from reference measuring devices over some period of observation, these comparisons cannot serve as a complete functional definition. They cannot substitute for the physical acts themselves.

The functionality of computation can be completely captured by symbols. Computational function is thus freed from a particular material substrate. The functionalities of measurement and control, on the

other hand, cannot be so separated from their material substrates--they are completely dependent upon the particular physical nature of the material structures which implement the functions.

Measurements and controls cannot be fully characterized within the observational frame in which they appear. An exhaustive description of these processes would necessitate that they be reduced to computations, as in an account of measurement using dynamical theory. But such accounts necessitate changing the observational frame so that many more (micro-) distinctions would be made for each existing distinction. The states of the new frame are microstates of the old frame; the states of the old frame are macrostates of the new frame.

> The second general remark is this: by definition, we cannot know the domain of stability of a particular modelling relation *from within the context of the modelling relation itself*. The bifurcation between system behavior and model behavior which we have described is itself an emergent novelty from the standpoint of the simple model; the basis for it has by definition been abstracted away in the very act of formulating the model. In order to *predict* this domain of stability, we need in fact another description; i.e. another model, with respect to which the original model will bifurcate....All we can say in general is that the domain of stability of a given modelling relation is *finite;* but we cannot in general predict the magnitude of this domain except through the agency of another model. (Rosen, 1985, p.337)

When we cross from necessary to contingent relations, we lose the ability to completely describe a process such that it can be exactly replicated within that observational frame. Determinacy is definite, indeterminacy is indefinite. Relative to a particular frame, the set of all possible computations on a given finite set of observable states can be enumerated and circumscribed. However, the set of all possible measurements and controls is not expressible by relations between the states of this frame such that we could exactly replicate the measurements and controls. These relationships are summaarized in figure 12.4.

Definite-symbolic (Explicate)	**syntactic**	**computations**	Exact comparisons between all elements possible; Given definite nature of elements and finite numbers, set of possible symbol manipulations is closed.
Partially-definite	**semantic pragmatic**	**measurementscontrols constructionsselections**	Exact comparisons between measurements & controls only possible through symbolic part; Partially definite elements yield indefinite, open set of distinguishable possibilities.
Indefinite (Implicate)	**nonsymbolic**	**nonsymbolic interactions**	No exact comparisons are possible, so set elements cannot be distinguished at all; Completely indefinite elements yield no set, since no distinctions can be made.

Figure 12.4. Definiteness, indefiniteness, and closure.

Adaptivity, emergence, and functional open-endedness

These considerations of functional definition, of exact and inexact replicability, and of bear on the issue of *functional open-endedness*. Functional open-endedness here is defined as the inability to define and enumerate all the possible functions available to the device in question. *Functional closure*, on the other hand, is the ability to carry out this definition and enumeration.

For each formal device, one and only one input-output function is possible at any given time. This input-output function can only change by the intervention of an outside agent. Each formal-computational device has a single, fixed input-output behavior which is completely describable in terms of a function mapping the set of input states to the set of output states. The set of possible input-output mappings for a completely programmable, computation-universal, formal-computational device is a finite, closed set, due to the finiteness of the number of states of the observer-device combination. The observed behavior of every formal-computational device can be described completely in symbolic terms and can therefore be exactly replicated.

Formal-robotic devices augment formal-computational devices with fixed measurements and controls. The computational part itself has a closed set of possible behaviors for the reasons listed in the last paragraph. The processes of measurement and control are inherently not state-determined processes, and therefore cannot be completely described via symbolic operations. We cannot enumerate the set of all *possible* measurements and controls. As a result, the set of all possible measurements and controls is open-ended, and the set of all possible formal-robotic devices is open-ended.

However, each formal-robotic device by definition employs a finite number of stable sensors and effectors. Each sensor and effector implements a definite semantic relation with the world, although we will not necessarily know the exact nature of that relation. Because we assume them to be fixed relations, we can label each fixed measuring and control device and the device will have some overall observed behavior which is definite and stable. The stability of this relation can never be known exactly, and can only be roughly gauged by reference to other similarly calibrated sensors and effectors and through changes in observed performance. Since measurement and control processes are contingent and not necessary operations, the specific outcome of each measurement or control cannot be replicated from their observed state transition description. But in terms of a particular measurement outcome, the device's behavior and level of performance will be the same over the entire period of observation. In this sense, the performance of a formal-robotic device is also replicable within limits, but not exactly so. The semantics of formal-robotic devices are *relatively closed*, bounded as they are by the limits of distinguishability of different functions.

Adaptive devices are not also completely replicable in their performance because their internal syntactic structure is based upon experience. Given the device itself, we do not know *a priori* which syntactic, computational mapping will be optimal for a given environmental context. We would have the device generate real world performances and adapt its syntax accordingly in order to ascertain the optimal feature-action mapping. Although we could not predict the behavior of our adaptive device's computational part without allowing it to test alternatives via real world performances, the range of possible input-output functions for that part can be *circumscribed*, because the set of possible input-output mappings is finite. Adaptive devices are therefore *syntactically closed*. General evolutionary devices are a different matter entirely. Syntactic primitives can be added through physical construction, so the computational part is *syntactically open-ended*. We no longer assume that set of sensors and effectors for a given device is fixed. As in formal-robotic devices, the set of *possible* measurements and controls is not enumerable, because the measurement and control operations are only partially describable in symbolic terms. Here, however, we are not constraining our devices to have *indistinguishably similar* behavior, but we demand that they have *distinguishably different* behavior. While we cannot define the measurement function in terms of our observable state transitions, we can determine whether the function has changed by comparing it with the results from a reference measuring device. If the two devices disagree, or more dramatically, if the sensor of the evolutionary

device changes such that a new measurement state transition comes into being (e.g. before the measurement resulted in one of two symbol states, after it resulted in one of three symbol states), then the function has changed. As a result, evolutionary devices are, relative to devices with fixed semantics, *semantically open*. The set of possible semantic relations such devices can have is apparently unbounded.

Chapter 13 Recapitulations and implications

> This amounts to saying that *theory of knowledge* and *theory of life* seem to us inseparable. A theory of life that is not accompanied by a criticism of knowledge is obliged to accept, as they stand, the concepts which the understanding puts at its disposal: it can but enclose the facts, willing or not, in pre-existing frames which it regards as ultimate. It thus obtains a symbolism which is convenient, perhaps even necessary to positive science, but not a direct vision of its object. On the other hand, a theory of knowledge which does not replace the intellect in the general evolution of life will teach us neither how the frames of knowledge have been constructed not how we can enlarge or go beyond them. It is necessary that these two enquiries, theory of knowledge and theory of life, should join each other, and, by a circular process, push each other on unceasingly. (Bergson, 1911, pp. xxiii-xxiv)

Recapitulations

1) A biological semiotics is useful for understanding the origins of cognitive function in biological systems. The categories of syntactics, semantics, and pragmatics, suitably developed, are capable of capturing the essential relations of evolutionary functional adaptation.

2) This biological semiotics can be related to the modelling relation in science and to the design of adaptive devices. Semiotic analysis clearly distinguishes the types of functional relations which must be operative if our models and devices are to be useful to us.

3) The analytical apparatus of physics can be used to describe and clarify the action of symbols. Symbols can be seen as special kinds of physical constraints, ones which can be described in terms of rate-independent rules.

4) The demarcation of symbolic from nonsymbolic can be used to distinguish different types of transformations between the two realms. Computation can be seen as a rule-governed, coupling of rate-independent symbols to other rate-independent symbols. Measurements can be seen as the coupling of rate-dependent, law-governed processes to rate-independent symbols. Controls can be seen as the coupling of rate-independent symbols to rate-dependent law-governed processes. Nonsymbolic interaction can be seen as law-governed rate-dependent processes. These categories are extremely useful in discussing the physical basis of and differences between computation, measurement, control, and nonsymbolic interaction.

5) Systems-theoretic criteria can be constructed to unambiguously distinguish computations, measurements, controls, and nonsymbolic interactions within a fixed, well-defined observational frame. These functionalities can be classified through observation.

6) Devices utilizing symbols can be analyzed in terms of the structural and functional inter-relationships, the kinds of linkages between their measurement, computation, and control functionalities. The plasticity of structural interrelationships and their degree of dependence on their performance leads to a taxonomy of different types of devices.

7) Formal-computational devices (computers, formal systems) are completely symbolic in their operation. Once programmed, their internal structure remains functionally unchanged by their external environment. As a consequence, formal-computational devices are nonadaptive. As a result of their nonadaptive structure, formal-computational devices are incapable of generating new functional primitives.

8) Formal-robotic devices (fixed robots) are syntactically and semantically nonadaptive. While formal-robotic devices have inherent semantics by virtue of their sensors and effectors, the structure of their semantic and syntactic parts is fixed; hence they are completely nonadaptive.

9) Adaptive devices (perceptrons, neural nets, genetic classifiers) have adaptive syntactics (performance-dependent computational parts) but they are bounded by their fixed semantic parts (sensors/features & effectors/actions).

10) Evolutionary devices have adaptive semantics. General evolutionary devices have both adaptive semantics and adaptive syntactics. While there are many natural examples of evolutionary devices, only one type has ever been constructed explicitly for this purpose (Pask's electrochemical adaptive devices).

11) The capacities and limitations of the various device types can be assessed. The more adaptable and plastic a device is, the less specification a designer must do, but the more training the device must undergo before it is useful. Formal-computational and formal-robotic devices most reliably and most efficiently carry out a problem solution once that solution has been specified. Adaptive devices are capable of finding effective syntactic, feature-action mappings, once appropriate features and actions have been found. Training, however, takes many more iterations than straight computation. Evolutionary devices are capable of finding appropriate features and actions, but this process, being dependent upon physical construction, is a very slow one.

12) Three types of emergence can be distinguished: computational emergence, thermodynamic emergence, and emergence relative to a model. The behavior of the various kinds of adaptive devices can be related to types of emergence relative to a model. This observer-centered theory of emergence can be used to describe the acquisition of new cognitive functions and the consequent amplification of the observer.

13) The device types and their associated types of emergent behavior can be considered in terms of the relative open-endedness or closure of their functional repertoires. Purely syntactic operations, being completely symbolic and well-defined, have closed sets of possibilities. Mixed symbolic-nonsymbolic, semantic operations, being only partially well-defined, have open sets of possibilities. The functional repertoires of formal-computational, formal-robotic and adaptive devices are closed, while those of evolutionary devices are open.

Implications

1) Computer simulations are incapable of generating new primitive structures and functions by themselves (chapter 7, appendix 3). The best computers can do is to build up compound categories through the application of pre-existing categorical primitives and combinatorial rules. No computer simulation has the ability to expand its state space, no matter how large it is.

2) Computations cannot completely take the place of measurements or controls, except in completely formal realms. We cannot completely dispense with empirical observables in our scientific models. Purely computational strategies will not generate new semantic relations with the physical world. We cannot get emergent real world functions from a computer simulation alone. Computer simulations cannot create physical motions by the act of representing them, no matter how complex they are.

3) Real world functioning requires connection with the physical world. We need to recognize this explicitly. The addition of measurements and controls to computations, as in robots, allows for this connection. Measurements and controls need to be taken as seriously as computations. Robots are qualitatively distinct from computers. A robot is different from its computer simulation.

4) Adaptive robots are qualitatively distinct from fixed robots. Devices which alter their structure through their interactions with the world are capable of exceeding their designers' abilities (disproving Descartes' Dictum, chapter 8). Consequently, adaptive devices can be used to augment the cognitive powers of human beings. Syntactically adaptive robots (adaptive devices) gather information relative to fixed categories of perception and action. Semantically adaptive robots (evolutionary devices) find useful new categories of perception and action.

5) While a fixed robot can always be built to carry out the same performances as its adaptive counterpart at any point in time, a fixed robot does not learn, and therefore cannot improve its performance over time.

6) We already know, more or less, how to build effective syntactically adaptive devices (neural nets, genetic algorithms). We do not know, in general, how to build effective semantically adaptive devices, although one type has been built (Pask's electrochemical device) and many design guidelines are available to us from biology (e.g. DNA → folded protein, immune system design). There is no reason why such devices cannot be built.

7) Evolutionary devices are qualitatively different from any kinds of devices now in use. They would be useful in situations where we cannot foresee what features and controls we will need to solve a problem. They can be used to redefine features for adaptive classifiers, making it easier to partition the feature space in an effective way. Evolutionary devices in the form of artificial immune systems would be useful in the recognition of complex macromolecules and the development of pharmaceuticals. When we multiply the number of independent perspectives on the world at large, we can enrich our own vantage point through communication (syntactic linkage).

8) An evolutionary robotics research program integrating syntactic, semantic, and pragmatic considerations could give us 1) a comprehensive, evolutionary "theory of knowledge" and 2) a pragmatically-based theory of the origins of new functions (a "theory of life"). The linkage of syntactic, semantic, and pragmatic relations into a mutually constraining adaptive network would be the first step toward the construction of an entity capable of semantic-closure, of self-construction and self-definition. For the first time we would be constructing an artificial entity capable of epistemic autonomy, albeit of a very limited sort. We might then better understand the interplay of relations which govern our own semiosic processes.

Appendix 1 Formal systems and potential infinities

> Evidently, the category of number is wonderfully consistent and complete as long as it is applied to counting real apples, but it becomes paradoxical when it is extended to such things as infinite sets, which transcend our experience. (Gunter Stent, "Introduction and Overview" in Delbrück, 1986, p. 11).

In many discourses potential infinities and Gödelian impotency principles have played a role in justifying the infinite richness of formal systems, the unpredictability of their behavior, and consequently the separation of generative rules from that behavior. Often, however, conclusions derived for potentially infinite realms do not apply for finite realms. Too often valid generalizations about arbitrary Turing machines are uncritically transferred to generalizations about finite state machines. So ubiquitous and automatic is the identification of Turing machines and the computers we use that it is almost always forgotten that real computers are finite state devices and that generalizations about Turing machines may not hold.

Extreme caution must be used whenever results from potentially infinite systems are used to justify conclusions about finite systems; care must be taken to verify that the conclusions reached can actually be applied to physically realizable devices.

Infinities may be useful for mathematical imagination and conceptualization, and there is no reason to banish them from this domain, but, in the interest of clarity, it may be best to disallow all talk of potential and actual infinities when we address problems concerning physically realizable devices.

Infinities and physical realizability

Physical realizability precludes the possibility of infinite numbers of operations or infinite strings of symbols. Aristotle was the first to make the argument that infinite entities could not exist in nature:

> Again, nothing infinite can exist; and if it could at least the notion of infinity is not infinite, i.e. does not contain an infinite number of marks (Aristotle, Metaphysics, 994b27-28)

Many prominent mathematicians of subsequent centuries, such as Henri Poincaré, were to concur with Aristotle:

> For my part I think, and I am not alone in so thinking, that the important thing is never to introduce any entities but such as can be completely defined in a finite number of words. (Henri Poincaré, 1952, Science and Method, p. 45)

With the rise of meta-mathematics and Cantor's theory of infinite sets at the end of the nineteenth century, the question of infinities became a crucial foundational issue for mathematics. Could proofs themselves be infinite in length? Is finiteness of the proof procedure a necessity? In the midst of speculations of this sort, David Hilbert argued that a proof of infinite length is formally meaningless, since it cannot be verified:

> A careful reader will find that the literature of mathematics is glutted with inanities and absurdities which have had their source in the infinite. For example, we find writers insisting, as though it were a restrictive condition, that in rigorous mathematics only a *finite* number of deductions are admissible in a proof -- as if someone had succeeded in making an infinite number of them. (Hilbert, 1925, p.184)

Hilbert's argument also applies to a proof with a finite *but very large* number of deductions, if there are so many that the reliability of the deductive apparatus itself is called into question. As human beings we are limited in the number of symbols we can process without error. Modern digital computers are, of course, orders of magnitudes more reliable than we, but their fallibilities and capacities nevertheless determine the boundaries of what can be represented and effectively computed.

Even Hilbert, who had just declared that "No one shall drive us out of the paradise which Cantor has created for us" believed that the notion of physically realizable infinities is a fundamentally confused idea:

> We have already seen that the infinite is nowhere to be found in reality, no matter what experiences, observations, and knowledge are appealed to. Can thought about things be so much different from things? In short, can thought be so far removed from reality? Rather is it not clear that, when we think that we have encountered the infinite in some real sense, we have merely been seduced into thinking so by the fact that we often encounter extremely large and extremely small dimensions in reality?
>
> Does material logical deduction somehow deceive us or leave us in the lurch when we apply it to real things or events? No! Material logical deduction is indispensable. It deceives us only when we form arbitrary abstract definitions, especially those which involve infinitely many objects. In such cases we have illegitimately used material logical deduction, i.e. we have not paid sufficient attention to the preconditions necessary for its valid use. (Hilbert, 1925, p.191)

Hilbert's program

Hilbert's program was based upon the construction of formal processes out of material tokens ("extralogical concrete objects") and physical operations on them. While he believed infinite collections of such tokens were not physically realizable, he did believe that consistent finite representations of infinite entities could be constructed and implemented through finite sets of physical tokens.

> As a further precondition for using logical deduction and carrying out logical operations, something must be given in conception, viz., certain extralogical concrete objects which are intuited as directly experienced prior to all thinking. For logical deduction to be certain, we must be able to see every aspect of these objects, and their properties, differences, sequences, and contiguities must be given, together with the objects themselves, as something which cannot be reduced to something else and which requires no reduction. This is the basic philosophy which I find necessary, not just for mathematics, but for all scientific thinking, understanding, and communicating. The subject matter of mathematics is, in accordance with this theory, the concrete symbols themselves whose structure is immediately clear and recognizable. (Hilbert, 1925, p.192)

These concrete tokens or primitives would then be combined in various ways to construct more complicated relations between the combinations. Von Neumann, who was associated with Hilbert's formalist program neatly summarized the goals of the movement:

> The leading idea of Hilbert's theory of proof is that, even if the statements of classical mathematics should turn out to be false as to content, nevertheless, classical mathematics involves an internally closed procedure which operates according to fixed rules known to all mathematicians and which consists basically in constructing successively certain combinations of primitive symbols which are considered "correct" or "proved." ...We must regard classical mathematics as a combinatorial game played with the primitive symbols, and we must determine in a finitary combinatorial way to which combinations of primitive symbols the construction methods or "proofs" lead." (von Neumann, 1931)

The goal of formalist mathematics was to construct a system of operations on tokens which was internally consistent, that is, no combination of allowed operations should lead to a contradiction.

Potential infinities

It could be argued that Hilbert's program went aground in trying to assume that the natural numbers, potentially infinite entities, could be represented by finite numbers of concrete symbols. While he was very skeptical of actual infinities, he was not skeptical of potential ones. Not even the most skeptical mathematicians of the period, such as Kronecker and Poincaré, doubted the usefulness of indefinitely extendible constructions like the natural numbers (potential infinities), even if they vehemently denied the validity of the concept of the totality of natural numbers (actual infinities)(Bernays, 1925; Troelstra & van Dalen, 1988, chap. 1).

A notion of formality based upon physical realizability eventually must clash with even potentially infinite constructions. The problem is that eventually any existing symbol recognition, manipulation, and storage capacities will be overwhelmed by the sheer numbers of recognitions and distinguishable tokens needed if a potential infinity is allowed. As physically realized observers, we have only the means to distinguish a finite number of states. Physical realizability and therefore, consistency, can only be guaranteed for finite constructions within the bounds of our recognition, manipulation, and storage capacities.

An illustrative example is the "doubly recursive" Ackermann's function. The function is given in three lines:

(1) $f(0, n) = n + 1$
(2) $f(m, 0) = f(m - 1, 1)$
(3) $f(m, n) = f(m - 1, f(m, n - 1))$

> It thus appears that this function is effectively computable, and we can compute its value for any given pair of arguments.
>
> However, [Ackerman's function] is a most remarkable function, and perhaps we should examine more closely our use of the phrase 'we can compute.' To find, say f(2,2) is a trivial computation and f(3,3), (=61) can be computed in a fraction of a second on any modern computer.
>
> But f(4,4) is another story. No one knows with certainty at this time if the universe is finite or infinite. However, current theories with some reasonable evidence to support them ... seem to imply that the universe is finite. It is enormous, however, and we are fairly certain of the existence of many billions of galaxies, most of them containing many billions of stars. The total mass of this finite (if it is) cosmos has been estimated to include something like 10^{80} elemental particles.
>
> If every one of these particles were used in some way to represent one digit in the decimal representation of f(4,4), not only would they not be sufficient, but *we would not even be able to represent that number which is the number of digits in f(4,4)* (i.e.,$[\log_{10} f(4,4)]$). What meaning then is there in the statement that 'we can compute' this function? (Beckman, 1980, p.128)

Computability issues

To most observers at the time, including Hilbert himself, Gödel's incompleteness theorems (Gödel, 1931; Nagel and Newman, 1958; Beckman, 1980) demolished the Hilbertian program (see Dawson, 1988; Shanker, 1988). Even so, there have always been those who doubted its significance and applicability. Bertrand Russell, in a 1963 letter to Leon Henkin, seemed to have doubts:

> If you can spare the time, I should like to know, roughly, how, in your opinion, ordinary mathematics--or, indeed, any deductive system--is affected by Gödel's work. (Russell, quoted in Dawson, 1988, p.90)

In contrast to Russell's ambiguity, Wittgenstein's forceful rejection of Gödel's theorems was based on a systematic critique of the platonic methodological assumptions embedded in the theorems. These include the notion of mathematical truth outside of a proof procedure, the meaningfulness of contradictory statements, the very concept of a hidden contradiction, the usefulness to us of mathematical truths, and so on. The assessment of Wittgenstein's position on these matters (e.g., was he a strict finitist?) involves a complex, ongoing discourse (Wittgenstein, 1956; Kielkopf, 1970; Wright, 1980; Shanker, 1987, Shanker, 1988) not *immediately* relevant in this context. Whether or not Wittgenstein was a strict finitist, Gödel's theorems can be assailed from that point of view.

For our purposes here, it should be remembered that Gödel's theorems rest upon the assumption that there can be indefinitely long string lengths, potential infinities. If finite string lengths are assumed, then the resulting logical system is completely surveyable and its consistency can be verified by logico-combinatoric means. Turing machines have potentially infinite tapes in order to be able to represent the

natural numbers. For Turing machines with finite tapes, the related Halting Problem disappears completely, along with all other computability issues. All of Gödel's results accrue from the potentially infinite (and indefinite) nature of his subject matter. After Gödel, the choice facing mathematicians was to either abandon all hope of proving the consistency of arithmetic with natural numbers (and all equivalent systems based upon potential infinities) or to restrict themselves to finite constructions, which most saw as overly restrictive. Most mathematicians were loathe to abandon so much of the foundational work based on the intuitive consistency of arithmetic on natural numbers (Kline, 1980). Mathematics is about ideas and patterns of thought, after all, and not the meaningless manipulation of tokens. But to use the machinery of symbol manipulation for the purposes of calculation is a different matter altogether, one in which we need not be concerned with anything but the meaningless manipulation of tokens.

The irrelevance of infinities to real world computations

All of the symbol manipulating devices we have at our disposal are restricted to finite constructions. All symbolic manipulations that can be carried out physically by our computers involve finite numbers of symbol primitives and distinguishable states. To restrict ourselves to manipulating finite strings of symbols does not limit what calculations we can perform. For our purposes discussions involving potential and actual infinities in the number of primitives (alphabets) and states (string lengths) are completely irrelevant, *since these operations are not physically realizable*. In terms of physically performable computations, exclusion of potential and actual infinities has no effect.

The benefits of finiteness

Why bother to clear the discourse of these notions if they are irrelevant to our purposes? Excluding potential and actual infinities does have the beneficial effect of removing indefinite entities from the discourse, which often serve as refuge for confusion and, occasionally, outright mysticism (e.g., Rucker, 1982). It promotes conceptual hygiene.

In the presence of discussions involving infinities, it becomes rapidly forgotten that while "infinity" can be *represented* by finite strings of symbols, those infinite strings of symbols cannot actually be *constructed*. The *caveats* of Aristotle, Poincaré and Hilbert, cited above, are ignored. Why?

Two reasons can be offered.

First, the postulation of indefinitely long symbol strings and syntactic operations gives platonists a means of arguing that their syntactic operations are not closed and finite in nature. One finds this type of reasoning in defenses of computer creativity (e.g. Turing, 1950; Hofstadter, 1979, 1985; Laing, 1989). In the computationalist worldview there are no new primitives, so that whatever new structures arise, *must* arise through *extension*. In terms of symbol strings, this means ever longer string lengths. Open-endedness in a completely syntactic system relies on indefinite extension, whereas open-endedness in a semantic system relies on new semantic primitives and the indefinite nature of the individual semantic relations.

Second, the postulation of indefinitely long Turing tapes and pushdown stacks ensures a qualitative separation of string types. The second reason underlies the apparent motivation for Pylyshyn's insistence that the brain is a Turing machine rather than a finite state automaton (see appendix 2 for discussion).

Is English a potentially infinite language?

A particularly notable example of both motivations is the conflation of ideal, potentially infinite languages with natural ones:

> Chomsky points out that there is no limit to the possible length of grammatical sentences in English and argues that English sentences are organized in such a way that this is sufficient to rule out a finite-state machine as a source of all possible English text. But, can we really regard a sentence miles long as grammatical when we know darned

well that no one ever has or will produce such a sentence and that no one could understand it if it existed! (Pierce, 1961, p.112)

We see that Chomsky has laid out a plan for a grammar of English .... The processes allowed in this grammar cannot be carried out by a finite-state machine, but they can be carried out by a more general machine called a Turing Machine, which is a finite state machine plus an infinitely long tape on which symbols can be written and from which symbols can be read or erased.

We should note, however, if we arbitrarily impose some bound on the length of a sentence, even if we limit the length to 1,000 or 1 million words, then Chomsky's grammar does correspond to a finite-state machine. The imposition of such a limit seems very reasonable in a practical way. (Pierce, 1961, pp. 114-115)

A great deal of the reason that Chomsky's language-strings are potentially infinite lies with a commonly held distinction between finite state machines, indefinitely extendible pushdown automata, and Turing machines. From the point of view of their total state descriptions, these distinctions are arbitrary labellings of particular total device states, as long as all pushdown stacks and Turing tapes are finite. There is no essential difference in the descriptions, but some descriptions are more useful for particular purposes. For those working in linguistics, it is important to separate input strings from machine instruction strings (the tape state - machine state distinction) and these strings from substrings to be recognized and to be inserted. The potential extendability of each of these types of strings ensures that they will remain separate in the description, resistant to a total state description which would submerge the distinction.

The *ideological* role of potential infinities in Chomskian linguistics is to argue for an indefinitely rich language based on syntactic open-endedness, but real human speakers can only remember on the order of 30 words at a time in a given sentence. As it happens, the open-ended nature of English arises not because of indefinitely long string lengths, but because of the creation of new words (as well as new letters and phonemes) and, most importantly, the creation of new word meanings by the speakers themselves. These creative processes come about through the actions of physical, biological creatures, human beings, who interact with the world and each other in both symbolic and nonsymbolic ways. Neither of these creative processes, as we shall see below, can be purely formal in character.

Fortunately, even the syntactic generativity of the idealized English that linguists study does not depend upon the unbounded nature of the Turing machine tapes, since the same generative algorithms can be implemented, albeit with speed and storage limits, by finite state automata. Consequently, the explanatory success of Chomskian linguistics is not dependent upon the unbounded machines its interpretational ideology requires, even if the unboundedness assumption yields a more convenient formalism for many linguistics. We can rest assured that our limited capacity brains and computers are not disqualified from English competency because of their finiteness.

Appendix 2: Do programs have inherent semantics?

The debate between the programs-are-pure-syntax view and programs-have-semantics view turns on the interpretation of what syntactics and semantics are about. How would we distinguish procedures with syntactic and semantic components from those which are pure syntax? How would we determine whether a symbol was "interpreted" or "uninterpreted" in an unambiguous way? This dissertation has argued that it *is* possible to discern whether a symbol is connected to the external material world, whether it has a stable "external" semantics. However, the relations of "internal semantics" or "logical semantics" seem to be completely syntactic in character, *being completely reducible to rule-governed relations between symbols. Is there a justification for making this distinction, or is it only a matter of notation?* Many fundamental distinctions in computational psychology, artificial intelligence, and artificial life depend upon the status of "internal semantics," so it seems useful to analyze its basis in two very influential works by Alan Newell and Zenon Pylyshyn.

If we examine closely the rationale that Newell (1980) makes for distinguishing between a semantically laden "symbol-level" and a purely syntactic manipulation of tokens, we see that it relies upon the distinction between machine specification and input. Newell is arguing that this distinction is essential and that it demarcates the tokens involved by their respective functions within the device (some code for machine instructions, some are raw input).

> A machine is defined to be a system that has a specific determined behavior as a function of its input. By definition, therefore, it is not possible for a given machine to obtain even *two* different behaviors, much less any behavior. The solution adopted is to decompose the input into two parts (or aspects): one part (the *instruction*) being taken to determine which input-output function is to be exhibited by the second part (the *input-proper*) along with the output. This decomposition can be done in any fashion -- for instance, by a separate input channel or by time (input prior to a starting signal being instruction, afterward being input proper). This seems like an innocent arrangement, especially since the input-proper may still be as open as desired (e.g. all future behavior). However, it constitutes a genuine limitation on the structure of the system. For instance, the instruction must have enough capacity to specify all the alternative functions. (If the instruction to a machine consists only of the setting of a single binary toggle switch, then the machine cannot exhibit three different input-output behaviors.) Most important, the basic decomposition into two parts has far reaching consequences -- it guarantees the existence of symbols. (Newell, 1980, pp.148-149)

There is no doubt that the tokens manipulated by the computer can be so divided into tokens which code for machine instructions and those which code for the input proper. The question is whether this distinction is essential:

> I am not inclined to accept, for example, [Newell's] characterization of a computer system level in general. While it is undeniable that many different -- radically different -- accounts of computational devices find common use, I don't understand what it is to say, for example, that a level 'really exists,' rather than being a point of view. Usually there are theoretical frameworks under which we describe computational devices; in one sense these are viewpoints, but that fact doesn't make the objects viewed from those viewpoints any less *real,* or descriptions in the viewpoint's terms any less *true*. I may describe the Nahanni River Canyon as a trap for paddlers' souls, and as a glacial scar; both statements may be true, without challenging the unproblematic existence of a river. (Smith, 1986, p.42)

As was argued in chapter 7 in the discussion of computational universality, all the various descriptions of the machine (global machine states, inputs and machine state-transition rules, inputs and machine state-transition rules and programs, etc...) are formally equivalent and interchangeable. Newell uses the purported essential nature of the distinction to demarcate between "symbols" having "internal semantics" and tokens which have only syntactic properties, but this distinction does not have the irreducibility claimed for it.

Pylyshyn at least recognizes the difficulties in grounding a theory on this distinction. He writes,

> It appears that we need to appeal to the content of representations in order to both account for certain kinds of generalizations and to give a principled account of certain constraints and capacities, such as those embodied in the principle of rationality. It also appears that, in a certain sense, it may be possible to accommodate this need within a functional framework by providing an interpretation of inputs and outputs and of at least some states of the model. Doing so, however, does not solve the entire problem, for it might still be argued that such interpretations are not the model's interpretations, that they are merely attributions made by the theorist. It does matter to us whether the states are representational for the model or merely for us. This is a question of whether the semantics of the states are original--as our thoughts presumably are--or whether they are derived as with the case with books. The meanings books have are in a derived sense; they mean something *only for us* because their authors intend them to have certain meanings when they write them. In cognitive science, however we want something stronger than derived semantics, inasmuch as we want to explain the system's behavior by appealing to the content of its representations.
>
> A dilemma is presented here. As we have already noted, the semantics of representations cannot literally cause a system to behave the way it does; only the material form of the representation is causally efficacious. (Pylyshyn, 1984, p.39)

Even given this lucid appraisal of the problem, Pylyshyn goes on to appeal to symbolic codes in the computation of numerical functions as a means of solving the dilemma. Although he doesn't frame it this way, in effect he says that we select the machine codes needed for a particular function we have in mind. This still doesn't solve the problem of semantic-laden representations and pure syntactics. Finally he gives up entirely by saying,

> As dyed-in-the-wool realists, we propose to do as the next stage exactly what solid-state physicists do when they find that their postulating certain unobservables provides a coherent account of a set of phenomena: we conclude that the codes are "psychologically real," that the brain is the kind of system that processes such codes and that the codes do in fact have a semantic content. (Pylyshyn, 1984, p. 40)

Asserting that the distinction of semantic-laden vs. purely syntactic symbol manipulations is equivalent to "postulating certain observables" effectively concedes the argument to those who hold that distinction to be in the mind of the theorist.

Why go through all of this agony? Pylyshyn wants to ground an autonomous "semantic"-laden "symbolic level" on the same distinction between state transition specifications and machine inputs. He goes through great lengths to argue that finite-state automata descriptions are fundamentally distinct from Turing machine descriptions of the same device behavior (pp.68-74) because in the Turing machine description there is a separation of "machine states" from the "tape states." Pylyshyn wants to make a corresponding separation between the "functional architecture" and cognitive "representations." In finite state automata it is immediately apparent that all of the different state labellings and overall descriptions are formally equivalent and interchangeable, but in the Turing machine the tape is indefinitely extendible, so the machine states and the tape states cannot be as readily merged into the same notation, since one is finite and definite and the other is potentially infinite and indefinite. This motivation is readily apparent in Pylyshyn's tenacious retention of the distinction in the face of the obviously finite capacity of any Turing machine in the brain:

> The difference between Turing machines and finite-state automata might seem to be merely that one is infinite and the other finite; but that isn't so .... It is true that the class of functions that can be computed by the two machines depends on the unboundedness of the Turing-machine tape. As far as the relevance of the notion of the Turing machine to cognition is concerned, however, the potentially infinite length of the Turing-machine tape serves to force a kind of qualitative organization on the process. (Pylyshyn, 1984, p.71)

This passage is a prime example of the covert use of potential infinities to attempt to enforce essential distinctions that do not apply to finite state machines (see app. 1). But a *finite* tape Turing machine and finite state automata have formally equivalent descriptions, so Pylyshyn is forced into the position of arguing for neural analogues for Turing machine control boxes and tapes. R.J. Nelson has criticized

Pylyshyn on these very grounds, that the distinction between Turing machines and finite state automata is not a functional one. He has also questioned the applicability of unbounded tapes for describing biological organisms: "Where is the organism's or person's tape? Is the tape potentially infinite in at least two dimensions? Is there more than one tape? What is the square on a person's tape, and what size is it?" (Nelson, 1987, p.407).

The collapse of the distinctions upon which the programs-have-semantics rests has implications for *strong artificial intelligence* and analogously, the strong artificial life viewpoint, as Pylyshyn explicitly recognizes:

> The cognitive theory would be gratuitous, or at best weakly equivalent or "mere mimicry," if the ascription of some particular representational content to the states of the model were not *warranted*. The particular interpretation placed on the states, however, appears to be extrinsic to the model, inasmuch as the model would behave exactly in the same way if some other interpretation had been placed on them. This observation is what has led some (for example, Searle, 1980) to claim that functional models (at least of the computational type) do not actually contain representations, that it is we, the *theorists*, who for reasons of convenience, hold that functional states in the computer model have representational content--and that adds up to what Searle (1980) calls *weak artificial intelligence,* that is, a functionalist account in which formal analogues "stand in" for but neither have nor explain the representational (or intentional) contents of mental states. (Pylyshyn, 1984, p.43)

One has to respect the intellectual honesty and clarity of Pylyshyn's presentation of the opposing view. For artificial life, the collapse of the programs-have-semantics view means that the computer simulations of artificial life are best regarded as just that, *simulations*, rather than *realizations,* of living organisms.

Appendix 3: Evolutionary simulations and emergence

Many strategies have been proposed to make evolutionary computer simulations emergent, but none so far have succeeded (Dewdney, 1988). Some of the ones proposed at various moments during the First Artificial Life Conference, September, 1987, were: implement simulations of morphological evolution, add morphogenetic rules, add meta-rules for recognizing "new" configurations, add chaotic and/or stochastic processes, make the environment more complex, make the fitness function "implicit," use bottom-up cellular automata simulations to represent hierarchical levels, use neural nets for the perceptual apparatus of simulated organisms, put connectionist networks on the "front end" of evolutionary simulations.

Each of these strategies falls prey to the static nature of simulations as formal-computational devices. As a result none of them, when conceived in purely computational terms, breaks out of the formal notational-specificational trap.

The problem of generating novelty in an evolutionary simulation

One sees the limitations of formal representations most sharply if one tries to simulate the emergence of new morphological structures in biological evolution. After attempting enough of these simulations, one begins to realize that whatever structures are generated have to be part of the simulating notation from the beginning; they must be represented in the model from the start. All structures which are represented in any model are necessarily composed of combinations of the symbol primitives of the model. No new primitives are ever created *de novo* in a computer simulation in the way that new functions are created in a biological system. In a computer simulation one cannot generate fundamentally novel primitives for the same reasons that a formal-computational system is incapable of creating new semantic or syntactic relations (chap. 7). The best one can do is to generate unexpected combinations of existing primitives, unanticipated behavior within completely anticipated categories. *All* evolutionary simulations, including those presented at the conference, necessarily exhibit this fundamental limitation common to all formal models.

Robert Rosen (1973) has pointed out that structural evolutionary models can be grouped into two categories, *transformation theories*, where all morphologies are parameterized from the beginning, as in D'Arcy Thompson (1917) and *emergence theories*, where simple elements with inherent dynamical properties interact to produce "emergent" aggregates. Of transformation theories Rosen says,

> 1. Transformation Theories. Underlying all theories of this type is the view that every biological species can be represented as a point in an appropriate kind of space; this space can be coordinatized by a sufficiently comprehensive collection of morphological characteristics, each of which takes on a definite value for a particular species. (Rosen, 1973, p. 113)

In transformation theories an evolutionary trajectory consists of the successive points in the morphological space which are traversed. All possible combinations of the primitive characteristics are already encoded into this space, so there is no building up of combinations of these primitives over time. Rosen cites two problems in these types of theories. The first is that "the individual states ("species") which occur in such theories do not represent the states of a real system, and the trajectories are not parameterized by real time" (Rosen, 1973, p. 113). These models do not describe real time-dependent behavior. The Dawkins Blind Watchmaker Program, discussed in the next section, is an example of such a transformation theory.

The limitations of the transformation theories flow from their static morphology space:

> Another difficulty ... is that it is assumed that all the morphological characteristics which co-ordinatize the space involved are known in advance, and that evolutionary developments are represented simply by a modification of the

> numerical values of these characteristics. This kind of view simply leaves no room for the generation of novel functions and structures which we recognize as the most fundamental and challenging aspect of evolution. (Rosen, 1973, p.114)

Emergence theories, on the other hand, start out with simple elements, combining them as the simulation progresses into aggregates. Cellular automata simulations, evolutionary simulations representing organisms as aggregates of interacting parts, and ecological simulations portraying ecosystems as aggregates of interacting organisms can all be seen as emergence theories in Rosen's terms:

> The viewpoint taken here is essentially that of descriptive chemistry; we start with a population of simple elements of several different kinds, which can interact to generate more and more complex structures. These structures may have properties completely different in kind from those of the elements of which they are composed, but which can be understood in terms of the properties of the elements themselves and the rules governing their interaction. (Rosen, 1973, p. 114)

In formal terms, both of these types of models turn out to be equivalent and closed. Both types of theories fail to create new state variables, to enlarge their state spaces, to create unanticipated possibilities.

> ...in formal terms, the "emergent" theories are no more emergent than the transformation theories are. For the dynamical description of states always remains at a low level of system description; e.g. in den [sic] OPARIN theory it remains at the level of the values assigned to the concentrations of chemical species in the "soup." There is no point at which we can identify functional aggregates of greater complexity, except by assigning a concentration number to them. That is, there is no way we can pass to descriptions of of aggregates which function as "individuals" at a higher level. (where by such an "individual" we would mean a pattern of temporal change in certain state variables which remain strongly correlated over long periods of time.) The problem can be graphically stated as follows: how would one go about finding whether there was an amoeba in a pot filled with amoeba constituents, when the only information which can be obtained about the system is the change of concentration of the constituents in time? Obviously that information is not sufficient to find whether there is an amoeba in the pot or not; what is necessary is new information, pertaining to the properties of a functional individual at a higher level of system description. This is precisely what is missing from the "emergent" theories; they talk about emergence only at a single level of system activity and system description; they do not generate new state variables in an effective way. (Rosen, 1973, p.114)

This is the same situation that we analyzed in chapter 4 regarding the nonderivability of higher order, nonholonomic constraints from the equations of motion. One might argue that in the course of a cellular automata simulation or evolutionary simulation that "higher level patterns" emerge, but these are distinctions generated by the human observer and not by the simulation itself (see chapter 12 for more discussion). Rosen concludes that "both the transformation theories and the emergence theories fail to incorporate, let alone account for, the generation of evolutionary novelties" (p.134). What is missing from these types of theories is *biological function:* "no dynamical model of evolution can have any claim to validity unless it incorporates the idea of biological function" (p.115). Rosen points out that both transformation and emergence theories are *structural theories*, rather than functional ones. I have, of course, throughout this dissertation argued for an understanding of symbols in biological organisms based on their syntactic, semantic, and pragmatic functions. I would argue that it is the process of encoding the properties of organisms into a computer simulation that necessarily reduces rate-dependent (and therefore at least partially implicate) functions to rate-independent, symbolic, structures. Computer simulations necessarily have this structural character.

We are now in a position to critically examine various specific break-out strategies which have been suggested, while retaining the above perspectives in mind. The criticisms of each approach should not be construed as criticisms of the particular models exemplifying the approach inasmuch as very few, if any, evolutionary simulations have been designed to grapple with this problem of fundamental novelty.

"Generating new primitives" or "achieving open-ended function creation" have not been stated goals of any of the simulations at the Artificial Life conference, so they should not be judged by such criteria.

There are hopeful signs that the limitations of simulations are beginning to be recognized. Packard (1989) describes his experiences with a simulation of bugs wandering about a 512 by 512 lattice searching for food and reproducing,

> There is one major aspect of biological evolution that cannot be captured by the model as it is presently described: in biological evolution, the biosphere seems to explore a constantly expanding universe of forms with new types of functionality and interactions emerging constantly; in the present model, the universe of forms available to the bugs is highly restricted, (a point ... in a two-dimensional space), and is currently specified *a priori* as a feature of the model. The primary generalization of the model currently being explored is to create an open space of forms in which the bugs may evolve. (Packard, 1989, p. 147)

He suggests two major strategies, making the bugs of his simulation less sophisticated, so they could evolve various strategies for detecting food and moving around. These would, of course, have to be programmed into the simulation, so although he would be dealing with a larger space of possible behaviors, the space of possibilities would still be closed. The second strategy is to increase the sophistication of the bugs: "the bugs could be allowed to interact, and could have an open space of possible interactions" including sexual reproduction, symbiosis, and predator-prey relationships. Again, a more complex simulation is programmed, enlarging the morphological state space and interactional alternatives, but the space is still closed. We need to think more strategically, in the manner of Rosen(1973) and Pattee(1973): how would we get the more complex simulation from the simpler one without allowing for all the elements of the more complex one in our program. I do not believe this can be done within a computer simulation, but this is the goal that those who believe in its possibility should aim for.

Adding morphogenetic rules to evolutionary simulations

What about adding variable rules for the morphogenesis of artificial organisms? At the conference Richard Dawkins presented an adaptive program for generating graphic representations of morphologies via a string-directed pattern language (Dawkins, 1989; Dewdney, 1988). The adaptive, selective part is qualitatively distinct from a purely computational evolutionary simulation because it is contingent upon the autonomous, unrepresented actions of the human selection agent. The pattern grammar, however, would be one major implementation route for morphogenetic processes in conventional evolutionary simulations. In Rosen's terms, this pattern grammar is a transformation theory.

Dawkins states his intentions at the beginning:

> With a program called the *Blind Watchmaker*, I created a world of two dimensional artificial organisms on the computer screen. ...I called them biomorphs. My main objective in designing *Blind Watchmaker* was to reduce to the barest minimum the extent to which I designed biomorphs. I wanted as much as possible of the biology of the biomorphs to *emerge*. All that I would design would be the conditions--ideally very simple conditions--under which they might emerge. The process of emergence was to be the Darwinian process of random mutation followed by nonrandom survival. Once a Darwinian process gets going in a world, it has an open-ended power to generate surprising consequences: us, for example. (Dawkins, 1989, p. 201)

As clever, well designed, and valuable as these sorts of programs are in allowing us to steer ourselves through a very complex pattern space, it is not hard to recognize the limits of a given pattern grammar, even one as rich as Dawkin's. *Everything which can happen in this evolutionary space is within the bounds of the pattern notation. We have no way of expanding the space of possible patterns by creating new pattern primitives.* Dawkins himself recognizes some of the limitations, "...I am fairly sure that there are *some* kinds of shapes that the Blind Watchmaker can never breed" (p.216), but he goes no

further. Perhaps because of the visual, concrete nature of morphologies, it becomes clear very early on that there can be no universal pattern notation. While morphogenetic rules are indispensable for more realistic representations of biological organisms and the incorporation of alternate morphological possibilities, such rules must be prespecified and operate on finite sets of represented morphological properties. There is no way out of this dilemma as long as only computations are involved. The set of morphological alternatives will still be closed, albeit potentially very large.

Adding rules to modify rules of interaction

What about adding rules to modify other rules of interaction between parts of organisms and between the organisms themselves? This was one strategy advocated by Packard(1989) in opening up his world of hungry, reproducing bugs. The strategy of adding, rules for changing state transition rules over the course of a computation, underlies many of the attempts to solve this problem of fixed boundaries. Adding these meta-rules will result in more complicated state transition rules, but because these rules must also be specified at the beginning of the simulation, so they appear to the designer as yet more state transition rules. One just has a more complicated interactions within the same variables and variable states. No new variables accrue from the process and the closed nature of the simulation is not affected.

The reason meta-rules seem like an answer is that a good many computer programmers make the conceptual fallacy, one which is pervasive in artificial intelligence, that our interpretation of the symbols being manipulated somehow changes the nature of that manipulation. A module or variable named "phenotype" is thought to actually be a phenotype, rather than a symbolic representation of a phenotype. This is equivalent to saying that a digital representation of an analog process is in fact analog because of what it represents, or that the mathematics of continuous processes (e.g. differential equations) actually involves manipulating continuous entities. There is a tendency to separate the meta-rules from the "local" rules of interaction because our interpretation of their role in the simulation is different.

From the point of view of formality and the replicability of results, meta-rules are indistinguishable from any other state transition rules. If we take the formal-computational device and treat it as any other deterministic physical system, we can measure its initial state (inputs) and we have access to its state transition rules (program). If we adopt the state transition rules of the program as our predictive algorithm, the final state of the device can be predicted just as for any other deterministic physical process. Thus, meta-rules will not affect the replicability of the simulation nor our ability to predict the final state. It may be that there is no shorter predictive algorithm that will completely predict the final state, but, for the purposes of replicability and predictability this is irrelevant (appendix 4). If it is possible to generate the behavior in a formal-computational device given the constraints of time, memory, and reliability, it is possible to replicate that behavior in a predictive algorithm.

Complexifying the simulated environment

The complexity of biological organisms reflects the complexity of their environments. A population of organisms in an evolutionary simulation will only evolve up to the limits of the detail of the environmental and organismic representations. In artificial worlds these environmental and organismic representations, together with rules for how the organisms are allowed to interact with each other and their environment must be completely specified. If some artificial organisms are to die in the simulation, the criteria for survival must be encoded into the program, either explicitly or implicitly. This is often done with an explicit fitness criterion or it can be handled implicitly, as a function of state variables of organisms (e.g. as in the "food-states" of the bugs in Packard, 1989). Formally, these two programming strategies are equivalent.

Even if one avoids using explicitly defined fitness functions, instead embedding survival criteria into the states of organisms themselves (e.g. adequate nutrient levels, avoidance of predators), the model still

has an explicit set of criteria which could be rewritten into an explicit fitness function doing exactly the same thing. At some point in the program, a decision is made as to which artificial organisms will survive and which ones will die, and the variables on which this decision is based can be traced back through the program. *This is still explicitly programmed.* While this could be a very laborious process and the resulting explicit survival function might be as complicated as the implicitly programmed version, there is no reason that an explicit survival function couldn't be derived through a rewriting of the rules of a given evolutionary simulation.

As Pattee pointed out at the conference, any formal representation of the environment is necessarily a fixed, finite, and impoverished one (Pattee, 1989). Artificial environments are inherently different from real environments in their relation to us, because we do not have to program a real environment in order for it to operate in a certain way. The real environment has autonomous dynamics relative to us. This lack of autonomous activity produces a problem of specification for us. Everything which is to happen in our artificial world must be allowed for in advance. If an artificial organism is to fly, either the capacity for flight must be represented explicitly as a variable state (e.g. FLIGHT=True) or implicitly as some combination of variable states (weight, body configuration and surface area, musculature, metabolic state, motivational state, that it's not constrained by a predator's jaws, etc). More variables and their interpretations can always be added by the designer, but this doesn't solve the problem of creating new properties which are not combinations of the existing ones. The strategy of adding micro-interactions, of going down a level to increase morphological and interactional alternatives just pushes the problem downwards a level. Whether the micro-processes involve morphogenetic rules or molecular dynamics, the micro-processes still have to be explicitly included in the program and the simulation will be closed at that level of detail. The simulation still will not lead to new variables or, of itself, to new interpretations. The simulation is still completely dependent upon its programmer(s) for its primitive representational categories.

Even if "event" detection modules are devised to scan the simulation looking for interesting behavior, as in Lenat's mathematics discovering program (Lenat, 1978, 1982; see also chap.7) and in the "shadow world" of Hogeweg (1989), the features which those modules utilize must be allowed for at the start by the programmer. No new primitive features are created during the operation of these programs which were not foreseen in some way by the programmer. What comes out of these simulations are combinations of prespecified features which were already available to the programmer. In Lady Lovelace's words, their "province is to assist us in making *available* what we are already acquainted with (Lady Lovelace, from Baum, 1986, p.82).

Adding chaotic and stochastic processes to evolutionary interactions

First, representing indeterministic or stochastic processes through pseudorandom or chaotic computations does not change the deterministic nature of the formal-computational device or the replicability of the simulation. There is a strong tendency to attribute to representations of a process the properties of that process. Indefinite entities must still be represented by definite, unambiguous symbols. Fuzzy, stochastic, continuous, and infinite entities must still be represented by crisp, deterministically calculated, discrete, finite strings of symbols. The difference is only in the user's interpretation of what they mean. Second, if genuinely random processes are incorporated, then the device will no longer be a formal-computational device, because the relation of the designer to the device would be fundamentally different. Depending upon the relationship of the randomness to the structure of the device, it might possibly make the device appear to be a formal-robotic one by making some computational state transitions appear as measurement and/or control transformations.

In either case it would still be possible to circumscribe the behaviors of the device, since adding random or chaotically generated state transitions would not affect the state space in which the (apparently) random transition occurred.

Chaotic computational processes do raise the apparent complexity of the simulation's behavior, in terms of the complexity of the algorithm needed to replicate it, but this really has nothing to do with emergence relative to a designer who has complete knowledge of the initial state and the state transition rules. Even if the only effective means of predicting exactly what will happen is to run the simulation itself, that does not mean that new categories have been formed. The behavior of the system is still circumscribable, still expressible in the original notation of possible outcomes.

What raising algorithmic complexity does is to make the simulation's behavior more difficult to comprehend to an outside observer lacking complete information about the system. It raises the level of surprise for that observer, but relative to that observer what is happening is no longer a formal computation, since initial states and state transition rules are not completely known by the observer and hence replicable. Understandably, there seems to be a strong tendency to attribute an objective novelty to newly discovered, unpredictable, ambiguous or poorly understood processes (e.g. chaos, fractals, fuzzy sets, connectionism, neural nets). Basic notions of emergence and open-endedness, which underlie the alternatives posed in this paper, are also subject to these tendencies when they are not clearly defined.

Emergence in cellular automata: in the machine or in the observer?

Cellular automata have also been suggested as another means of transcending the problem of specification. Cellular automata would create higher level structures out of the initial primitives, which would then act in some way upon other structures to produce new interactions and possible structures (see discussion in chap. 12). Upon seeing these spectacular displays (e.g. Tamayo & Hartman, 1989), it is hard to dismiss the notion that somehow there should be a way of using these structures generated by cellular automata to create new unanticipated interactions in an open-ended way. Much of the reason we tend to ascribe open-ended behavior and emergent properties to cellular automata simulations is due to the ease with which we as observers can recognize higher order patterns, such as cycles, waves and self-sustaining, complex configurations. These simulations and the insights which they facilitate are indispensable for the understanding and design of complex systems. When viewing a simulation, however, it is *we* who are generating novel interpretations of the simulation, not the simulation itself. It is *we* who recognize the higher order structures. The primitive categories of the simulation have not changed. No new interactions which were not in the initial simulation have not been created. It is extremely useful that such simulations provoke us to think in new terms; that's what make cellular automata so exciting, but if we are interested in designing devices which can construct new distinctions for themselves, we must look outside of purely symbolic transformations altogether.

Are massively parallel machines a solution?

What types of behavior would be necessary to generate fundamental emergence in a computer? What is needed is for there to be physically coherent, *rate-dependent dynamics* of interaction *which are not specified explicitly* by the designer from the start, *which could spontaneously come into play* during the course of a simulation. These would have to be rules or dynamics which could change without invoking yet another prespecified rule. An example of this would be if the global behavior of a cellular automaton began to modify or constrain the local rules of interaction, without the global-local interactions being specified by the simulation's designer from the start. In the Game of Life, the local rules of a node might spontaneously change based upon whether it was in an area completely enclosed by "on" nodes. New patterns would arise. The simulation would then be open-ended since new stable symbol primitives, and hierarchies of interaction could arise which were not preprogrammed. The behavior of the system would no longer be reducible to the prespecified local rules of interaction. *It should be clear we are no longer talking about computers in any normal sense of the word* (or in the restricted sense of chapter 7). Suddenly the state transition rules would be ambiguous and the input-

output function would be rendered indeterminate. Results would not be replicable, and the central, defining characteristic of formality would be lost. Such a system would be nonprogrammable in Conrad's sense (Conrad, 1983).

Those connectionist, neural net, catalytic network, and simulated-annealing approaches which do not rely completely on rule-based interactions (whether microscopic or macroscopic), and hence are not completely programmable, are positive steps in the direction of harnessing implicate dynamics towards more open-ended behavior. Most studies of these non-rule-governed, dynamical relaxations, however, are rule-governed computer *simulations*, completely within the digital, computational domain. As simulations, present parallel computational networks stand as symbolic, discrete, *representations* of continuous, law-driven, analog, physical interactions. To the extent that these networks operate within completely symbolic categories and within completely encoded environments these devices will be trapped within static state spaces. Therefore, if new interactions and open-ended behavior are desired, the analog, physical network processes can not just be simulated, they must be physically implemented, or no new interactions will come into play. Consequently, in this view there is a sharp distinction between those research strategies which see the computed simulations as algorithmic ends-in-themselves and those which see the simulations as a representational tool for understanding and eventually using mixed digital-analog physical processes themselves for functionalities other than computations.

Appendix 4 Prediction in physics and computer science

In chapter 12 it was argued that the computationalist sense of the word "emergence" shifts the meaning of the term away from its original meaning in order to fit the capabilities of computers and formal systems, to syntacticize it. We have seen this happen once before with the syntacticizing redefinition of the term "semantics" to refer to relations between symbols (chapter 2). Similarly, the concept of "prediction" has been altered in this discourse.

In the physical sciences, prediction generally connotes that one has a means of foretelling the course of subsequent observed events given some initial observation. As in the modelling frameworks of Hertz and Rosen discussed in chapter 4, the act of prediction involves two processes: measurements to establish the initial state and computations to execute the formal model, which we shall call the predictive algorithm. For the prediction to be valid, "the necessary consequents of the images in thought [the model] are always images of the necessary consequents in nature of the things pictured" (Hertz, 1956, p. 1). In other words, the subsequent observed behavior of the physical system must correspond to the results of the predictive algorithm, given the initial measured state. An empirically adequate predictive algorithm has complete correspondence with the behavior of the observed system; it captures, or replicates, this behavior completely.

If we find an observational frame and a predictive algorithm which captures the subsequent behavior of the observed system given observation of its initial state, then we would say that the algorithm *predicts* the behavior of the physical system. This usage is obvious when we are talking about physical systems, say, balls on inclined planes or electrical voltages and currents, but objections are often raised regarding the predictability of computer simulations.

Some of these objections flow from the purported unpredictability of the results of certain uncomputable functions. As we took great pains to point out in chapter 7 and appendix 1, the non-computability of these functions depends upon having unlimited numbers of potential states, and therefore these noncomputable functions cannot be realized by physical systems in the first place, due to their unbounded nature. Computability constraints always lie outside of physical realizability constraints.

Objections flowing from the physical limitations of the predictive algorithm can also be met by noting that a deterministic physical system capable of high relative complexity of behavior can also be used to implement the predictive algorithm. If we were going to predict the behavior of a connectionist machine, our predictive algorithm itself could be implemented via another connectionist machine.

Objections to the isomorphism between the two machines is often made, because if you have to carry out the computation in order to predict it, in what sense is this a prediction? You are merely replicating the computation. If we are predicting the behavior of a large, complex physical system, in most cases we will not be able to forecast the behavior of our predictive algorithm. We run the algorithm once with a particular set of parameters, and there is no point in doing it again if we are reasonably confident in the reliability of our computational instrument. We will not be surprised if all subsequent runs are the same.

Prediction in computer science generally implies "predictable by means of a shorter algorithm," which substitutes a criterion of formal complexity for replicability. Mathematical chaos, for example, is thought to be "unpredictable" in this sense, because in many cases there is no simpler algorithm for capturing the behavior than the one which generated it. Note, however, this is a different sense of the word from the sense used in physics. If we had access to the chaos-generating algorithm and we incorporated this into our predictive model, we could *predict* the behavior of the computer in the physics sense of the word, but this would not be a *prediction* in the computer science sense of the word, since we used the same algorithm. Real chaos, the kind of *observed physical behavior* of stock markets, weather systems, and turbulent flows needs to be distinguished from mathematical chaos, because such

phenomena *are* effectively unpredictable in the physics sense, due to their extreme sensitivity to perturbations of all kinds.

It is understandable why the computer science sense has the meaning it does; if it meant the same as it does in physics, *all algorithms* would be predictable and there wouldn't be much point in using the word at all. In its sense of "being able to predict using a shorter algorithm" it serves a much more useful purpose in pointing to an important means of increasing algorithmic efficiency.

An example of this shift of word meanings lies in the strong artificial life argument that the phenotype of artificial organisms is not "predictable" from its genotype (Langton, 1989). Here "predictable" means, in the computer science sense of the word, that a *shorter* algorithm for computing the genotype may not be available. Unfortunately, it can also connote that the mapping cannot be reliably replicated or logically deduced by an observer, even one with a paper and pencil and a good deal of of time. This latter conclusion, of course, would be incorrect.

Linearity, nonlinearity, predictability, and replicability

Linearity is another term which has acquired a new meaning in the context of computer simulations. Langton (1989) gives a fairly straightforward definition of linearity as it is commonly understood in physics and engineering:

> ...*linear systems* are those for which the behavior of the whole is just the sum of the behavior of its parts, while for *nonlinear systems*, the behavior of the whole is more than the *sum* of its parts.
>
> Linear systems obey the *superposition principle*. We can break up complicated linear systems into simpler constituent parts, and analyze these parts *independently*. Once we have reached an understanding of the parts in isolation, we can achieve a full understanding of the whole system by *composing* our understanding of the isolated parts. This is the key feature of linear systems: by studying the parts in isolation, we can learn everything we need to know about the complete system.
>
> This is not possible for nonlinear systems ... even if we could reach a complete understanding of the parts in isolation, we would not be able to combine our understanding of the individual parts into an understanding of the whole system. The key feature of nonlinear systems is that their primary behaviors of interest are properties of the interactions between parts, rather than being properties of the parts themselves, and these interaction-based properties necessarily disappear when the parts are studied independently. (Langton, 1989, p. 41)

Despite this definition, Langton claims that cellular automata and artificial life simulations exhibit nonlinear behavior.

> With bottom-up specifications, the system computes the local, nonlinear interactions explicitly, and the global behavior--which was implicit in the local rules--emerges spontaneously without being treated explicitly. (Langton, 1989, p.42)

Given his definition, however, automata are decomposable, linear systems *par excellence*. In what sense is a computer program's behavior "more than the sum" of its modules' behaviors? How do the input-output functions of program modules change when they are connected in different ways?

If we have an cellular automaton simulation and we analyze the state-transition behavior of each node, that state-transition behavior will not change as a result of being connected to many other nodes. The behavior of a network of these nodes can be *logically deduced* from the state-transition behavior of the individual nodes. The simplest way to go about finding the necessary logical consequences of a particular set of nodes connected in a particular way is generally to run a simulation of them. *Having to simulate the system to understand it does not change the linearity of the relationship or the reducibility of the behavior of the whole to the behavior of the parts*. Once we have simulated the behavior of the connected nodes, we are in a position to understand the system as a whole from the behavior of the parts.

In fact, the "linearity" of formal-computational devices contrasts markedly with the "nonlinearity" of biological systems, and this is yet another way of saying that computational devices have no emergent properties, while biological systems exhibit many emergent properties:

> Now biological systems are typically organized in a manner like that of the three body system: they resist physical decomposition into subsystems which possess the same dynamical properties in isolation that they did when connected into the system. Thus, although certain of their properties can be studied by such physical decompositions, many others (including some of the most important) cannot be. For instance, all of the hierarchical ones cannot be; we cannot physically separate the cellular level of biological organization from higher or lower levels. It is this fact, more than any other, which is responsible for the 'counter-intuitive' properties of organized systems, and in particular for their capabilities for exhibiting completely unexpected properties (the so-called 'emergent' properties) which seem unpredictable on the basis of analysis predicated only on the properties of physically separated and described subsystems. (Rosen, 1974, p. 172)

There is a semantic sleight-of-hand going on throughout this discourse. Langton's stated meaning of "linear" is that the behavior of the aggregate system can be "understood" from an understanding of the behavior of its components. It may be what he means is *context-dependent*, since when he discusses *simple linear growth* (p. 26), he is talking about *context-free* formal languages.

Similar semantic tangles surround Langton's use of predictability in his (literal) analogy of a Turing machine and the genotype-phenotype distinction. He wants to demonstrate that the artificial phenotype is not predictable from the artificial genotype, which would be an important requirement for their distinctness. If artificial phenotypes are logically derivable from artificial genotypes, it would be difficult to maintain that the genotype/phenotype distinction, an essential one to biologists, was anything but an arbitrary relation between symbols, a construction of the theorist (see appendix 2). Artificial life in its strong form claims to realize life, but if the genotype/phenotype distinction cannot be replicated in a convincing way, the strong interpretation is in trouble. Unfortunately, Langton's argument turns on the dual meaning of predictability. This is in a section labelled "Unpredictability of PTYPE from GTYPE" (Langton, 1989, p.23). The Turing machine's transition table is regarded as a genotype (GTYPE), while the computation is regarded as the phenotype (PTYPE). He says, "...we cannot predict the PTYPES which will emerge from specific GTYPES given specific initial structures." The essential nature of the genotype-phenotype distinction in artificial life simulations and consequently, the fate of *strong artificial life* hangs on this point. He continues,

> As discussed previously, we know that it is impossible in the general case to determine *any* nontrivial property of the future behavior of a sufficiently powerful computer from a mere inspection of its program and initial state alone. (Langton, 1989, p.23)

A general text on automata theory is referenced without any specific page numbers, so it isn't clear exactly what is meant by "sufficiently powerful computer" or "nontrivial property." If "unpredictable" means "not being able to determine a nontrivial property by mere inspection and initial state," then by this usage, *nearly all calculations would be unpredictable*. Not *unpredicted*, but *unpredictable*. He continues,

> From this we can deduce that in the general case it will not be possible to determine, by inspection alone, any nontrivial feature of the PTYPE that will emerge from a given GTYPE in the context of a particular initial configuration. In general, the only way to find out anything about the PTYPE is to start the system up and watch what happens as the PTYPE develops under the control of the GTYPE. (Langton, 1989, p.23)

In these cases, by "predictability," Langton apparently means being able to see the result immediately, "by inspection alone," without having to run out a complex calculation. In fact this interpretation of "predictability" and the related "understandability" is also the only one which makes his account of

"linearity" coherent. Using this interpretation, yes, most computer simulations are "nonlinear," and "unpredictable." It should be clear by now, however, that in terms of logical derivability or replicability of result or the conventional meaning of predictability, the artificial life phenotype is *completely predictable* from the artificial genotype, given the initial configuration and the algorithm.

Note that this is a much different definition of predictability and understandability than is found in in physics, where even the simplest "predictions" may involve calculations that cannot be done by visual inspection alone. In that domain predictability connotes an ability to replicate the observed behavior of a system by means of a formal model. In a computer simulation we *always* have a formal model of the system, the computer program itself. Occam's razor notwithstanding, in physics the complexity of the predictive algorithm relative to the complexity of the physical systems is not relevant because generally the complexity of the physical system is not known. Even when applied to cases like weather prediction, where the predictive model or simulation is so complex that the phenomenon happens faster than the computation, the conventional meaning of the word is unchanged.

Appendix 5 Von Neumann and biological computationalism

John von Neumann is often adopted by advocates of the information-processing model who believe that his theory supports a computationalist view of biological symbols. Von Neumann, after all, is widely regarded as "the father of the electronic digital computer." He was an early advocate of computer simulation and the use of "the axiomatic method" to study the formal properties of complex systems. His discussion of problems of self-complexification of automata and the role of descriptions in self-reproducing systems were so conceptually clear and concise that they have not been surpassed, even today. His self-reproducing cellular automaton is the prototype for a large number of artificial life simulations.

Clearly von Neumann should be a formidable ally in the formation and definition of artificial life as an autonomous scientific research domain. While von Neumann's perspective strongly supports the *questions* posed by artificial life, his writings manifestly resist interpreting him as supporting the biological computationalist position.

Superficially, one can see him as a computationalist if one conceives of his self-reproducing automata primarily as formal structures rather than real physical devices. It should be puzzling to those advocates that von Neumann calls Turing machines "fictitious mechanisms" and "axiomatic paper automata" (von Neumann, 1949, p.446) and that he began his work on self-reproducing automata trying to outline how a real, physical device, his kinematic automaton, would work.

A number of other factors militate against simplistic interpretations of von Neumann as a biological computationalist. von Neumann's proof in quantum mechanics regarding the measurement problem implies that measurements cannot be computed, that the two processes are complementary, irreducible. This position is diametrically opposed to the computationalist-realist position that such a distinction between observations and formal computations is irrelevant. von Neumann was also associated with Hilbert's program to ground the philosophy of mathematical symbols in the physical realm of perceptible differences. By no stretch of the imagination can we regard von Neumann as a platonist of any sort.

In the biological realm von Neumann repeatedly emphasized the "mixed digital-analog" character of organic processes:

> The human organism is not a digital organ either, though one part of it, the nervous system, is essentially digital. Almost all the nervous stimuli end in organs which are not digital, such as a contracting muscle or an organ which causes secretions to produce a chemical. To control the production of a chemical and rely on the diffusion rate of a chemical is to employ a much more sophisticated analog procedure than we ever use in analog computing machines. The most important loops in the human system are of this nature. A system of nervous stimuli goes through a complicated network of nerves and then controls the operation of what is essentially a chemical factory. The chemicals are distributed by a very complicated hydrodynamical system, which is completely analog. These chemicals produce nervous stimuli which travel in a digital manner through the nervous system. There are loops where this change from digital to analog occurs several times. So the human organism is essentially a mixed system. But this does not decrease the necessity for understanding the digital part of it. (von Neumann, 1949, p.472)

At the very least, his writings reflect a very deep concern with the digital-analog distinction, (in this dissertation it has been called the symbolic-nonsymbolic or symbol-matter distinction). It is also clear von Neumann saw the genotype-phenotype relation in terms of a digital (computational) genotype and an analog (non-computational) phenotype:

> Now, in this context, [of mixed digital-analog processes] the genetic phenomena play an especially typical role. The genes themselves are clearly parts of a digital system of components. Their effects, however, consist of stimulating the formation of specific chemicals, namely of definite enzymes that are characteristic of the gene involved, and therefore, belong in the analog area. (von Neumann, 1958, p.69)

Many biological computationalists fail to see the essential distinction that the simulation obliterates. They tend not to perceive the simulation of the genotype-phenotype distinction as problematic (e.g. Langton, 1989, Laing, 1989). As was argued in chapter 3, a formal simulation is a completely digital process, and one cannot get truly analog processes in a digital simulation. To realize a genotype-phenotype distinction one must build a physical device with both analog and digital modes. If one takes the analog-digital distinction seriously, one has to *actually build* a device or at least think in terms of a real device, as von Neumann originally did with his kinematic self-reproducing automaton. This may be why he labelled his cellular automaton as a "formalistic" representation of the real thing.

It would be very difficult to reconcile von Neumann's view of the genotype-phenotype distinction with the computationalist belief that all the relevant biological processes are digital in character and therefore can be fully captured by means of formal simulations. von Neumann had very substantial doubts that this is not the case, that there will always be unexplained "analog" interactions going on which are outside the model once a phenomenon is formalized. If the analog interactions outside the axiomatization are important for the capturing of the interesting behavior of the system, then a digital axiomatization will destroy "the more important half of the problem."

> By axiomatizing automata in this manner, one has thrown half of the problem out the window, and it may be the more important half. One has resigned oneself not to explain how these parts are made up of real things, specifically how these parts are made up of actual elementary particles, or even of higher chemical molecules....These things will not be explained; we will simply assume that elementary parts with certain properties exist. The question one can hope to answer, or at least investigate, is: What principles are involved in organizing these elementary parts into functioning organisms, what are the traits of such organisms, and what are the essential quantitative characteristics of these organisms? I will discuss the matter entirely from this limited point of view. (von Neumann, 1949, p.480)

A very similar passage, where von Neumann discusses the axiomatization of neurons, is more indicative:

> ... They [McCulloch and Pitts] said they did not want to axiomatize the neuron as it actually exists, but they wanted to axiomatize an ideal neuron, which is much simpler than the real one. They believed that the extremely amputated, simplified, idealized object which they axiomatized possessed the essential traits of the neuron, and that all else are incidental complications, which in a first analysis are better forgotten. Now, I am quite sure that it will be a long time before the point is agreed to by everybody, if ever; namely whether or not what one overlooks in this simplification had really better be forgotten or not. (von Neumann, 1949, p.447)

Von Neumann is wondering whether the analog interactions in neurons are important or not for real brains. In the next paragraph he goes on to describe the McCulloch-Pitts notation:

> The definition of what we call a neuron is this, One should perhaps call it a formal neuron, because it certainly is not the real thing, although it has a number of the essential traits of the real thing. ... [the formal neuron's] main trait is that it can excite other neurons. Somewhere at the end of an involved network of neurons the excited neuron excites something which is not a neuron. For instance, it excites a muscle, which then produces physical motion; or it excites a gland which can produce a secretion, in which case you get a chemical change. So, the ultimate output of the excited state really produces phenomena which fall outside our present treatment. These phenomena will, for the sake of the present discussion, be entirely disregarded. (von Neumann, 1949, p.447)

Here the analog outputs of the muscles and glands are outside of the completely digital treatment of the neuron. This dissertation contends that the "more important half of this problem" lies at the digital-analog interface, not within the digital domain. It at this digital-analog interface that all the semantic primitives are fixed; it is here that the symbols of the digital realm get connected to the world at large. It is in the analog part of this interface that emergent behavior arises.

The above quotes are not an isolated lines of von Neumann's thoughts. In *The Computer and the Brain*, he discusses characteristics of neurons which would not be expressible in computational terms (pp. 53-56). He concludes:

> On all these matters certain (more or less incomplete) bodies of observation exist, and they all indicate that the individual neuron may be--at least in suitable special situations--a much more complicated mechanism than the dogmatic description in terms of stimulus-response, following the simple patterns of elementary logical operations, can express. (von Neumann, 1958, p.56)

von Neumann goes on to discuss the possible role of mixed digital-analog processes in the nervous system:

> It is conceivable that in the essentially digitally organized nervous system the complexities referred to play an analog or at least a 'mixed' role. It has been suggested by such mechanisms more recondite over-all electrical effects might influence the functioning of the nervous system. It could be that in this way certain general electrical potentials play an important role and that the system responds to the solutions of potential theoretical problems in toto, problems which are less immediate and elementary than what one normally describes by the digital criteria, stimulation criteria, etc. (von Neumann, 1958, pp.58-59)

The idea of analog interactions changing the structure of the digital system seems to be hinted here. If so, the resulting perspective is much closer to the one advocated in this paper, than it is to a strong biological computationalist view.

At the very least the structure of von Neumann's considerations should support a worldview broader in scope that of computationalism, one in which analog and mixed digital-analog processes play as important a role as digital-computational ones.

References

Abraham, Ralph (1976) Vibrations and the realization of form. In: Evolution and Consciousness: Human Systems in Transition. E Jantsch & CH Waddington, eds. Addison-Wesley, Reading, MA.

Abraham, R & CD Shaw (1984) Dynamics--the Geometry of Behavior. Vols I-III. Aerial Press, Santa Cruz, CA.

Ackley, David H, Geoffrey E Hinton, & Terrence J Sejnowski (1985) A learning algorithm for Boltzmann machines. *Cognitive Science* 9: 147-169, reprinted in: Anderson & Rosenfeld (1988).

Anderson, James A. & Edward Rosenfeld, eds. (1988) Neurocomputing: Foundations of Research. MIT Press, Cambridge, MA.

Aristotle. The Basic Works of Aristotle. Richard McKeon, ed. Random House, New York, 1941.

Arthur, Wallace (1985) Mechanisms of Morphological Evolution. John
Wiley & Sons, New York.

Ashby, W Ross (1952) Design for a Brain. Chapman & Hall, London.

(1956) An Introduction to Cybernetics Chapman & Hall, London.

(1962) Principles of the self-organizing system. In: Modern Systems Research for the Social Scientist: A Sourcebook. W Buckley, ed. Aldine, Chicago, 1968, reprinted from Principles of Self-Organization. H von Foerster & G Zopf, eds, Pergamon Press, New York, 1962.

(1965) Analysis of the system to be modelled. In: Mechanisms of Intelligence: Ross Ashby's Writings on Cybernetics. Roger Conant, ed. Intersystems Publications, Salinas, CA, 1981.

Axelrod, Robert (1981) The evolution of cooperation. *Science* 211:1390-1396 (27 March, 1981), reprinted in: Evolution Now: A Century After Darwin. John Maynard Smith, ed. WH Freeman & Co., San Francisco.

Barto, AG & RS Sutton (1981) Goal seeking components for adaptive intelligence: An initial assessment. *Air Force Wright Aeronautical Laboratories / Avionics Laboratory Technical Report AFWAL-TR-81-1070,* Wright-Patterson AFB, Ohio.

Barto, AG, RS Sutton & CW Anderson (1983) Neuronlike adaptive elements that can solve difficult learning problems. *IEEE Transactions on Systems, Man and Cybernetics* SMC-13 (5): 835-846, also reprinted in: Neurocomputing: Foundations of Research. James A. Anderson & Edward Rosenfeld, eds., MIT Press, Cambridge, MA.

Baum, Joan (1986) The Calculating Passion of Ada Byron. The Shoe String Press, Hamden, CT.

Beckman, ES (1980) Mathematical Foundations of Programming. Addison-Wesley, Reading, MA.

Beer, Stafford (1984) Personal communication. American Society for Cybernetics Annual Meeting, November 1984, Philadelphia.

Bergson, Henri (1911) Creative Evolution. Random House, 1946.

Beurton, P (1981) Organismic evolution and subject-object dialectics. In: The Philosophy of Evolution. UJ Jenson & R Harre, eds. The Harvester Press, Brighton, Sussex, UK.

Boden, Margaret (1981) Philosophical Psychology and Computational Models. Chapter 4. The case for a cognitive biology. Cornell University Press, Ithaca, NY.

(1988) Computer Models of the Mind. Cambridge University Press, Cambridge, England.

Bohm, David (1952) A suggested interpretation of the quantum theory in terms of "hidden" variables, I & II. *Physical Review* 85: 166-193. In: Wheeler & Zurek (1983).

(1957) Causality and Chance in Modern Physics. University of Pennsylvania Press, Philadelphia.

(1980) Wholeness and the Implicate Order. Routledge & Keegan Paul, London.

Bohr, Niels (1934a) Atomic Theory and the Description of Nature. Cambridge University Press, Cambridge, UK.

(1934b) Discussion with Einstein on epistemological problems in atomic physics. In: Atomic Physics and Human Knowledge. John Wiley, 1958. Reprinted by Ox Bow Press, Woodbridge, CT, 1987.

(1935a) Quantum mechanics and physical reality. *Nature* 136: 65. In: Wheeler & Zurek (1983).

(1935b) Can quantum-mechanical description of physical reality be considered complete? *Physical Review* 48:696-702. In Wheeler & Zurek (1983).

(1954) Unity of knowledge. In: Atomic Physics and Human Knowledge. John Wiley, 1958, reprinted by Ox Bow Press, Woodbridge, CT, 1987.

Bonner, John Tyler (1980) The Evolution of Culture in Animals. Princeton University Press, Princeton, NJ.

(1988) The Evolution of Complexity by Means of Natural Selection. Princeton University Press, Princeton, NJ.

Brooks, David & Wiley, EO (1986) Evolution and Entropy: Towaard a Unified Theory of Biology. University of Chicago Press, Chicago.

Brooks, Rodney A (1986) A robust layered control system for a mobile robot. *IEEE J. Robotics & Automation* RA-2(1): 1-10.

(1987) Achieving artificial intelligence through building robots. MIT Artificial Intelligence Lab paper, 545 Technology Square, Cambridge, MA, 02139.

Campbell, DT (1987) Evolutionary epistemology. Blind variation, selective retention in creative thought as in other knowledge processes. In: Radnitzky & Bartley (1987).

Campbell, R (1985) An organizational interpretation of evolution. In: Depew & Weber (1985).

Carello, Claudia, MT Turvey, Peter N Kugler & Robert E Shaw (1984) Inadequacies of the computer metaphor. In: Handbook of Cognitive Neuroscience. M Gazzaniga, ed., Plenum Press, New York.

Cariani, Peter & Narendra S Goel (1985) On the computation of the tertiary structure of globular proteins. IV. Incorporation of secondary structure information. *Bulletin ofMathematical Biology* 47(3): 367-407.

Carnap, Rudolf (1928) The Logical Structure of the World: Pseudoproblems in Philosophy. (Der Logische Aufbau der Welt.) Rolf A. George, trans. University of California Press, Berkeley, CA, 1967.

Cassirer, Ernst (1944) An Essay on Man. Yale University Press, New Haven, CT. Reprinted by Bantam Books, Toronto, 1970. Page numbers refer to the reprint.

(1955) The Philosophy of Symbolic Forms. Volume 1: Language. R Manheim, trans. Yale University Press, New Haven, CT.

(1957) The Philosophy of Symbolic Forms. Volume 3: The Phenomenology of Knowledge. R Manheim, trans. Yale University Press, New Haven, CT.

Cavallo, Roger E (1979) The Role of Systems Methodology in Social Sciences Research. Martinus Nijhoff Publishing, Hingham, MA.

Churcher, John (1982) Implications and applications of Piaget's sensorimotor concepts. In: Adaptive Control of Ill-Defined Systems. Oliver Selfridge, Edwina Rissland, and Michael Arbib, eds. Plenum Press, New York.

Churchland, Paul M (1985) The ontological status of observables: in praise of the superempirical virtues. In: Churchland & Hooker, eds. (1985).

Churchland, Paul M & Clifford A Hooker, eds. (1985) Images of Science: Essays on Realism and Empiricism, with a Reply from Bas C van Fraassen. University of Chicago Press, Chicago.

Churchman, C West (1971) The Design of Inquiring Systems: Basic Concepts of Systems and Organizations. Basic Books, New York.

Collier, John (1988) The dynamics of biological order. In: Weber, Depew & Smith, eds. (1988).

Conrad, Michael (1972) The limits of biological simulation. *Journal of Theoretical Biology* 45: 585-590.

(1974) Evolutionary learning circuits. *Journal of Theoretical Biology* 46:167-188.

(1983) Adaptability. Plenum Press, New York.

(1985) On design principles for a molecular computer. *Communications of the ACM* 28(5):464-480

(1986) What is the use of chaos? In: Holden, ed. (1986).

Dawkins, Richard (1987) The Blind Watchmaker. WW Norton, New York.

Dawson, John W (1988) The reception of Gödel's Incompleteness Theorems. In: Shanker (1988).

Denbigh, KG (1975) An Inventive Universe. George Braziller, New York.

Dennett, Daniel C (1984) Elbow Room. MIT Press, Cambridge, MA.

Depew, David D & Bruce Weber (1988) Consequences of nonequilibrium thermodynamics for the Darwinian tradition. In: Weber, Depew & Smith (1988).

Depew, David & Bruce Weber, eds. (1985) Evolution at a Crossroads: The New Biology and the New Philosophy of Science. MIT Press, Cambridge,MA.

Delbrück, Max (1986) Mind from Matter? An Essay on Evolutionary Epistemology. Blackwell Scientific Publications, Inc., Palo Alto, California.

Dewdney, AK (1988) Computer recreations: a blind watchmaker surveys the land of biomorphs. *Scientific American* 258 (2): 128-131, (Feb. 1988).

Dewey, John (1928) Human progress and social organization. In: The Philosophy of John Dewey. J Ratner, ed. Henry Holt & Co., New York.

Dreyfus, Hubert L. (1979) What Computers Can't Do. Harper & Row, New York.

(1981) From micro-worlds to knowledge representation: AI at an impasse. In: Mind Design: Philosophy, Psychology, Artificial Intelligence. J Haugeland, ed. MIT Press, Cambridge, MA.

(1986) Why computers may never think like people. *Technology Review* January 1986.

Dreyfus, Hubert L & Stuart E Dreyfus (1986) Mind over Machine: The Power of Human Intuition and Expertise in the Era of the Computer. The Free Press (MacMillan), New York.

(1987) How to stop worrying about the frame problem even though it's computationally insoluble. In: The Robot's Dilemma. Z Pylyshyn, ed. Ablex Publishing, Norwood, NJ.

(1988) Making a mind versus modelling the brain: artificial intelligence back at a branchpoint. *Daedelus* Winter, 1988.

Drexler, K Eric (1986) Engines of Creation. Doubleday, Garden City, New York.

(1989) Biological and nanomechanical systems: contrasts in evolutionary capacity. In: Langton, ed. (1989).

Eco, Umberto (1976) A Theory of Semiotics. Indiana University Press, Bloomington., Ind.

(1984) Semiotics and the Philosophy of Language. Indiana University Press, Bloomington, Indiana.

Edelman, Gerald M (1987) Neural Darwinism: The Theory of Neuronal Group Selection. Basic Books, New York.

Eigen, M & P Schuster (1978) The hypercycle, a principle of natural self-organization. Pt B. *Naturwissenschaften* 65:10-14.

(1979) The Hypercycle. Springer-Verlag, Berlin.

Feyerabend, Paul (1973) Against Method. NLB, London.

(1978) Science in a Free Society. NLB, London.

(1981a) Realism, Rationalism & Scientific Method. Volume 1: Philosophical Papers. Cambridge U. Press, Cambridge.

(1981b) Problems of Empiricism. Volume 2: Philosophical Papers. Cambridge University Press, Cambridge, UK.

(1987) Farewell to Reason. Verso, London.

Flanagan, Owen J (1984) The Science of the Mind. MIT Press, Cambridge, MA.

Fodor, Jerry (1975) The Language of Thought. Harvard University Press, Cambridge, MA.

(1980) Fixation of belief and concept acquisition. In: Piatelli-Palmarini, ed. (1980).

Fogel, LJ, AJ Owens, and MJ Walsh (1966) Artificial Intelligence Through Simulated Evolution. John Wiley, New York.

Frazzetta, TH (1975) Complex Adaptations in Evolving Populations. Sinauer Associates, Sunderland, Massachusetts.

Gause, Donald, and Gary Rogers (1976) Genetic pattern synthesis-an approach to man-machine symbiotic design. In: *Proceedings, Third European Meeting of Cybernetics and General SystemsTheory*, Vienna, Austria.

(1982) Cybernetics and artificial intelligence. In: Cybernetics: Theory and Application. R. Trappl, ed. Hemisphere Publishers, Washington, DC.

Gatlin, LL (1972) Information Theory and the Living System. Columbia University Press, New York.

Gibson, James J (1966) The Senses Considered as Perceptual Systems. Houghton-Mifflin, Boston.

(1982) Perceptual learning: differentiation or enrichment? Reasons for realism. In: Selected Essays of James J. Gibson. Edward Reed and Rebecca Jones, eds. Lawrence Erlbaum Associates, Hilldale, NJ, pp. 317-332, reprinted from *Psychological Review* 62: 32-41, 1955.

Gödel, Kurt (1931) On formally undecidable propositions of Principia Mathematica and related systems I. J van Heijenoort, trans. In: From Frege to Gödel. J. van Heijenoort, ed. Harvard University Press, Cambridge, MA, 1967.

Goldman, David E (1989) Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesly, Reading, MA.

Goodman, Nelson (1951) The Structure of Appearance. Third Edition. D. Reidel Publishing, Dordrecht-Holland, 1977.

(1956) A world of individuals. In: The Problem of Universals. Notre Dame University Press, Notre Dame, Indiana. Reprinted in: Philosophy of Mathematics. First Edition. P Benaceraff & H Putnam, eds. Prentice-Hall, Englewood Cliffs, NJ, 1964. Also reprinted in Goodman (1972).

(1963) The significance of Der Logische Aufbau der Welt. In: The Philosophy of Rudolf Carnap. PA Schilpp, ed., Cambridge University Press, La Salle, Illinois.

Goodman, Nelson (1972) Problems and Projects. Bobbs-Merrill, Indianapolis, Indiana.
(1976) Languages of Art. Hackett Publishing Co., Indianapolis, Indiana.
Goodwin, BC (1970) Biological stability. In: Towards a Theoretical Biology Volume 3. Drafts. CH Waddington, ed. Aldine, Chicago.
(1978) A cognitive view of biological process. *J. Social Biol. Struct.* 1:117-125.
(1982) Development and evolution. *Journal of Theoretical Biology* 97: 43-55.
Gould, Stephen Jay & Richard C Lewontin (1977) The spandrels of San Marco and the Panglossian paradigm: a critique of the adaptationist programme. In: Conceptual Issues in Evolutionary Biology: An Anthology, E Sober, ed. MIT Press, Cambridge, MA.
(1980) Is a new and general theory of evolution emerging? *Paleobiology* 6(1):119-130, reprinted in:Evolution Now: A Century After Darwin. John Maynard Smith, ed. WH Freeman & Co., San Francisco.
Greene, John C (1981) Science, Ideology, and World View: Essays in the History of Evolutionary Ideas. University of California Press, Berkeley, CA.
Grossberg, Stephen (1988) Nonlinear neural networks: principles, mechanisms and architectures. *Neural Networks* 1(1): 17-61.
Gunderson, Keith (1985) Mentality and Machines. Second edition. University of Minnesota Press, Minneapolis.
Habermas, Jurgen (1979) Communication and the Evolution of Society. Thomas McCarthy, trans. Beacon Press, Boston.
Haugeland, John (1980) Semantic engines: an introduction to mind design. In: Mind Design: Philosophy, Psychology, Artificial Intelligence. Haugeland, ed. MIT Press, Cambridge, MA.
(1985) Artificial Intelligence: The Very Idea. MIT Press, Cambridge, MA.
Hilbert, David (1925) On the infinite. In: Philosophy of Mathematics: Selected Readings. Second Edition. Paul Benacerraf & Hilary Putnam, eds., Cambridge University Press, Cambridge, England, 1983.
Hillis, Daniel (1988) Intelligence as an emergent behavior; or the songs of Eden. *Daedalus,* Winter, 1988,175-189.
Hirschorn, Larry (1984) Beyond Mechanization: Work and Technology in the Postindustrial Age. MIT Press, Cambridge.
Hofstadter, Douglas R. (1979) Gödel, Escher, Bach: an Eternal Golden Braid. Basic Books, New York.
(1982) Artificial Intelligence: subcognition as computation. Tech. Report No. 132, Computer Sci. Dept., Indiana U.
Hockett, CF (1958) A Course in Modern Linguistics. MacMilllan, New York.
Hogeweg, P. (1989) Mirror beyond mirror: puddles of life. In: Langton, ed. (1989).
Holland, John (1976) Adaptation in Natural and Artificial Systems University of Michigan Press, Ann Arbor.
Holden, AV, ed. (1986) Chaos. Manchester University Press, Manchester, England.
Holton, Gerald (1973) The roots of complementarity. In: Thematic Origins of Scientific Thought. Harvard University Press, Cambridge, MA, Revised ed.,1988.
Hopfield, JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *PNAS* 79: 2554-2558. Reprinted in Anderson & Rosenfeld, 1988.
Hopfield, JJ & David W Tank (1986) Computing with neural networks: a model. *Science* 233:625-633, (8 August, 1986).
Iberall, AS (1977) A field and circuit thermodynamics for integrative physiology. I. Introduction to the general notions. *Am. J. Physiol*. 233 (5): R171-R180.
Jackson, Phillip C. (1985) Introduction to Artificial Intelligence. Second edition. Dover, New York.
Janlert, Lars-Erik (1987) Modelling change--the frame problem. In: The Robot's Dilemma: The Frame Problem in Artificial Intelligence. Z Pylyshyn, ed. Ablex Publishing, Norwood, New Jersey.
James, William (1892) Psychology. The Briefer Course. Gordon Allport, ed. University of Notre Dame Press, Notre Dame, Indiana, 1985.
(1925) The Philosophy of William James. Horace M. Kallen, ed. Modern Library, New York.
(1968) The Writings of William James. John J. McDermott, ed., Modern Library, New York.
Jammer, Max (1974) The Philosophy of Quantum Mechanics. John Wiley. New York.
Kacser, H and R Beeby (1984) Evolution of catalytic proteins. *Journal of Molecular Evolution* 20:38-51.
Kauffman, Stuart (1984) Emergent properties in random complex automata. *Physica D* 10: 145-156.
(1985) Self-organization, selective adaptation, and its limits: a new pattern of inference in evolution and development. In: Depew & Weber (1985).

Kelly, Michael (1987) Natural and Artificial Symbol Systems: Construction and Computation in a Simple Organism. Ph.D.dissertation, Department of Systems Science, State University of New York at Binghamton, Binghamton, NY.

Kelso, JAS, JP Sholz & G Schöner (1988) Dynamics governs switching among patterns of coordination in biological movement. *Physics Letters A* 134(1): 8-12 (12 Dec. 1988).

Kielkopf, Charles F (1970) Strict Finitism. Mouton & Co., The Hague.

Kirkpatrick, S, CD Gelatt, & MP Vecchi (1983) Optimization by simulated annealing. *Science* 220: 671-680, also reprinted in: Neurocomputing: Foundations of Research. JA. Anderson & E Rosenfeld, eds., MIT Press, Cambridge, MA.

Klee, Robert L (1984) Micro-determinism and concepts of emergence. *Philosophy of Science* 51: 44-63.

Kline, Morris (1980) Mathematics: The Loss of Certainty. Oxford University Press, New York.

Klir, George J (1968) An Approach to General Systems Theory. Von Nostrand, New York.

Köhler, Wolfgang (1955) Direction of processes in living systems. *Scientific Monthly* 80 (I): 29-32. Reprinted in: The Selected Papers of Wolfgang Köhler. M Henle, ed. Liveright Press, New York.

Kohonen, Teuvo (1988) An introduction to neural computing. *Neural Networks* 1(1): 3-16.

Kugler, Peter N, JA Scott Kelso, MT Turvey (1980) On the concept of coordinative structures as dissipative structures: I. Theoretical lines of convergence. In: Tutorials in Motor Behavior. GE Stelmach & J Requin, eds. North-Holland, NY.

Lachman, Roy, Janet Lachman, and Earl C Butterfield (1979) Cognitive Psychology and Information Processing. Lawrence Erlbaum Associates, Hillsdale, NJ.

Laing, Richard (1989) Artificial organisms: history, problems, direction. In: Langton, ed. (1989).

Lakoff, George (1987) Women, Fire and Dangerous Things: What Categories Reveal about the Mind. U. Chicago, Chicago.

Langton, Christopher (1984) Self-reproduction in cellular automata. *Physica D* 10:135-144.

(1986) Studying artificial life with cellular automata. *Physica D* 22:120-149.

(1989) Artificial life. In: Langton, ed. (1989).

Langton, Christopher, ed. (1989) Artificial Life, Santa Fe Institute Studies in the Sciences of Complexity. Vol. 5. *(Proceedings of the First Conference on Artificial Life, Los Alamos, September, 1987.)* Addison-Wesley, Reading,MA.

Lenat, Douglas B (1978) The unbiquity of discovery. *Artificial Intelligence* 9: 257-285.

(1982) The nature of heuristics. *Artificial Intelligence* 19: 189-249.

Lettvin, JY, HR Maturana, WS McCulloch & WH Pitts (1959) What the frog's eye tells the frog's brain. Proceedings of the IRE Vol. 47, No. 11: 1940-1959 (November 1959). Reprinted in: Warren S. McCulloch Embodiments of Mind. MIT Press, Cambridge, MA, 1965, 1988.

Lilienfeld, Robert (1978) The Rise of Systems Theory: An Ideological Analysis. Wiley, New York.

Lindsay, Robert Bruce and Henry Margenau (1936) Foundations of Physics. Ox Bow Press, Woodbridge, Connecticut, 1981.

Lippmann, Richard P (1987) An introduction to computing with neural nets. *IEEE ASSP Magazine*, April, 1987, pp. 4-22.

Lloyd, GER (1968) Aristotle: The Growth and Structure of his Thought. Cambridge University Press, Cambridge, UK.

Lucas, JR (1961) Minds, machines and Gödel. In: Minds and Machines. AR Anderson, ed. Prentiss Hall, Englewood Cliffs NJ, 1964.

Lyons, John (1977) Semantics. Vol. 1.Cambridge University Press, Cambridge, UK.

MacKay, Donald M. (1969) Information, Mechanism and Meaning. MIT Press, Cambridge, MA.

MacKinnon, Edward (1985) Bohr on the foundations of quantum theory. In: Niels Bohr: A Centenary Volume. AP French and PJ Kennedy, eds. Harvard University Press, Cambridge, MA, pp.101-120.

Maruyama, Margaroh (1977) Heterogenistics: an epistemological restructuring of biological and social sciences. *Cybernetica* 20: 69-86.

Maturana, Humberto R. (1981) Autopoiesis. In: Autopoiesis: A Theory of Living Organization. Milan Zeleny, ed., North Holland, New York.

(1985) Biology of language: the epistemology of reality. In: Psychology and Biology of Language and Thought: Essays in honor of Eric Lenneberg. George Miller and Elizabeth Lenneberg, eds., Academic Press.

Maturana, Humberto R & Francesco J Varela (1987) The Tree of Knowledge. The Biological Roots of Human Understanding. Shambhala Press, Boston.

McEwan, John D. (1963) Anarchism and the cybernetics of self-organizing systems. In: A Decade of Anarchy (1961-1970). C Ward, ed. Freedom Press, London, 1987.

Mead, Carver & MA Mahowald (1988) A silicon model of early visual processing. *Neural Networks* 1(1): 91-97.

Michaels, Claire F & Claudia Carello (1981) Direct Perception. Prentice Hall, Englewood Cliffs, NJ.

Miller, James Grier (1978) Living Systems. McGraw-Hill, New York.

Minch, Eric (1987) The Representation of Hierarchical Structure in Evolving Networks. Ph.D. dissertation, Department of Systems Science, State University of New York at Binghamton.

Minsky, Marvin L (1967) Computation: Finite and Infinite Machines. Prentice-Hall, Englewood Cliffs, NJ.

Moravec, Hans (1988) Mind Children: The Future of Robot and Human Intelligence. Harvard University Press, Cambidge, MA.

Moray, Neville (1982) Humans and their relation to ill-defined systems. In: Adaptive Control of Ill-Defined Systems. Oliver G. Selfridge, Edwina L. Rissland, and Michael A. Arbib, eds. Plenum Press, New York.

Morgan, C Lloyd (1931) Emergent evolution. Henry Holt & Co., New York.

Morris, Charles (1946) Signs, Language, and Behavior. George Braziller, New York.

(1955) Foundations of the theory of signs. In: Foundations in the Unity of Science: Toward an International Encyclopedia of Unified Science. Volume 1, Nos. 1-10. O Neurath, R Carnap, C Morris, eds. University of Chicago Press, Chicago.

Morrison, Phillip and Emily Morrison, eds. (1961) Charles Babbage and His Calculating Engines, including Passages from The Life of a Philosopher and "Sketch of the Analytical Engine Invented by Charles Babbage by L. F. Menabrea" with "Notes upon the Memoir by the Translator, Ada Augusta, Countess of Lovelace." Dover, New York.

Mumford, Lewis (1966) The Myth of the Machine: Technics and Human Development.Harcourt, Brace, Jovanovich, New York.

Munévar, Gonzalo (1981) Radical Knowledge: A Philosophical Inquiry into the Nature and Limits of Science. Hackett Publishing, Indianapolis, Ind.

Murdoch, Dugald (1987) Neils Bohr's Philosophy of Physics. Cambridge University Press, Cambridge, UK.

Nagel, Ernest & James R Newman (1958) Gödel's Proof. New York University Press, New York.

Nilsson, Nils J (1965) Learning Machines: Foundations of Trainable Pattern-Classifying Machines. McGraw-Hill, New York.

Nelson, RJ (1987) Machine models for cognitive science. *Philosophy of Science* 54: 391-408.

Newell, Allen (1980) Physical Symbol Systems *Cognitive Science* 4(2):135-83.

(1986) The symbol level and the knowledge level. In: Meaning and Cognitive Structure: Issues in the Computational Theory of Mind. Z Pylyshyn & W Demonpoulos, eds. Ablex Publishing, Norwood, NJ.

Newell, Allen and Herbert Simon (1981) Computer science as empirical inquiry: symbols and search. In: Mind Design. John Haugeland, ed. Bradford Books, Montgomery, Vermont.

Oyama, Susan (1985) The Ontogeny of Information: Developmental Systems and Evolution. Cambridge University Press, Cambridge, UK.

Packard, Norman (1989) Intrinsic adaptation in a simple model for evolution. In: Langton, ed. (1989).

Pask, Gordon (1958a) The growth process inside the cybernetic machine. *Proceedings, Second Congress International Association of Cybernetics, Namur, Belgium*.

(1958b) Physical analogues to the growth of a concept. In: *Mechanization of Thought Processes: Proceedings of a Symposium held at the National Physical Laboratory on 24th, 25th, 26th and 27th of Nov., 1958*, H.M.S.O., London.

(1959) The natural history of networks. In: Self-Organizing Systems. MC Yovits & S Cameron, eds. Pergamon Press, New York, 1960. *(Proceedings of an Interdisciplinary Conference on Self-Organizing Systems, Chicago, May 5-6, 1959.)*

(1961) The cybernetics of evolutionary processes and of self-organizing systems.*Third Congress International Association of Cybernetics, Namur, Belgium.*

Pattee, Howard H. (1961) On the origin of macromolecular sequences. *Biophysical Journal* 1: 683-710.

(1968) The physical basis of coding and reliability in biological evolution. In: Towards a Theoretical Biology. 1.Prolegmena. CH Waddington, ed., Edinburgh University Press, Edinburgh.

(1969) How does a molecule become a message? *Developmental Biol.Supp*. 3: 1-16.

(1971) The problem of biological hierarchy. In: Towards a Theoretical Biology, 3. Drafts. CH Waddington, ed. Edinburgh University Press, Edinburgh.

(1972a) The nature of hierarchical controls in living matter. In: Foundations of Mathematical Biology, Vol. I, R Rosen, ed. Academic Press, New York.

(1972b) Laws and constraints, symbols and languages. In: Towards a Theoretical Biology. 4. Essays. CH Waddington, ed.Edinburgh University Press, Edinburgh.

(1972c) Physical problems of decision-making constraints. *International Journal of Neuroscience*, 3: 99-106.

(1973a) Physical problems in the origin of natural controls. In: Biogenesis, Evolution, Homeostasis. A Locker, ed., Pergamon Press, New York.

(1973b) Discrete and continuous processes in computers and brains. In: The Physics and Mathematics of the Nervous System. W. Guttinger & M. Conrad, eds. Springer-Verlag, New York.

(1973c) The physical basis of the origin of hierarchical control. In: Hierarchy Theory: The Challenge of Complex Systems. H Pattee, ed. George Braziller, New York.

(1974) The vital statistics of quantum mechanics. In : Irreversible Thermo-dynamics and the Origin of Life. GF Oster, IL Silver & GA Tobias, eds. Gordon & Breach, New York.

(1977) Dynamic and linguistic modes of complex systems. *International Journal of General Systems* 3:259-266.

(1979) The complemetarity principle and the origin of macromolecular information. *Biosystems* 11: 217-226.

(1980) Clues from molecular symbol systems. In: Signed and Spoken Language: Biological Constraints on Linguistic Form. U Bellugi & M Studdert-Kennedy, eds. Dahlem Konferenzen 1980 Weinheim: Verlag Chemie GmbH.

(1982) Cell psychology: an evolutionary view of the symbol-matter problem. *Cognition and Brain Theory* 5: 325-341.

(1985) Universal principles of measurement and language functions in evolving systems. In: Complexity, Language, and Life: Mathematical Approaches. J Casti & A Karlqvist, eds. Springer-Verlag, Berlin.

(1988) Instabilities and information in biological self-organization. In: Self-Organizing Systems: The Emergence of Order. FE Yates, ed. Plenum Press, New York.

(1989) The measurement problem in artificial world models. In: Langton, ed. (1989).

Piaget, Jean (1971) Biology and Knowledge. University of Chicago Press, Chicago.

(1976) Behavior and Evolution. Pantheon, New York.

(1980) In: Piatelli-Palmarini, ed. (1980).

Piatelli-Palmarini, Massimo (1980) How hard is the hard core of a scientific paradigm? In: Piatelli-Palmarini, ed. (1980).

Piatelli-Palmarini, Massimo, ed. (1980) Language and Learning. The Debate between Jean Piaget and Noam Chomsky. Harvard University Press, Cambridge, MA.

Pierce, JR (1961) Symbols, Signals and Noise. Harper & Bros., New York.

Poincaré, Henri (1913) The Foundations of Science: Science and Hypothesis, The Value of Science, Science and Method. George Brude Halsted, trans. The Science Press, Lancaster, PA, 1946.

Polyani, Michael (1964) Personal Knowledge. University of Chicago Press, Chicago.

Popper, Karl (1987) Natural selection and the emergence of mind. In: Radnitzky & Bartley (1987).

Porter, Arthur (1969) Cybernetics Simplified. Barnes & Noble, New York.

Prigogine, Ilya (1980) From Being to Becoming. WH Freeman San Francisco.

Prigogine, Ilya & Isabelle Stengers (1984) Order out of Chaos. Man's New Dialog with Nature. Bantam Books, New York.

Pylyshyn, ZW (1980) Computation and cognition: issues in the foundations of cognitive science.*Behav.Brain Sci.* 3:111-69.

(1984) Computation and Cognition. MIT Press, Cambridge, MA.

(1987) Preface. In: The Robot's Dilemma. Zenon W Pylyshyn, ed. Ablex Publishing, Norwood, NJ.

Radnitzky, Gerard & WW Bartley III, eds. (1987) Evolutionary Epistemology, Rationality and the Sociology of Knowledge. Open Court, La Salle, IL.

Riedl, Rupert (1978) Order in Living Organisms. RPS Jeffries, translator. John Wiley & Sons, Chichester, England.

Ritchie, GD & FK Hanna (1984) AM: a case study in AI methodology. *Artificial Intelligence* 23: 249-268.

Rorty, Richard (1979) Philosophy and the Mirror of Nature. Princeton University Press, Princeton, NJ.

(1985) Pragmatism and philosphy. In: After Philosophy: End or Transformation? K Baynes, J Bohman, & T McCarthy, eds. MIT Press, Cambridge, MA.

Rosen, Robert (1967) Optimality Principles in Biology. Butterworths, London.

(1969) Hierarchical organization in automata-theoretic models of biological systems. In: Hierarchical structures. LL Whyte, A Wilson & D Wilson, eds. Elsevier, New York.

(1970) Dynamical System Theory in Biology. Wiley, New York.

(1971) Some realizations of (M,R) systems and their interpretation. *Bulletin of Mathematical Biophysics* 33: 303-319.

(1973) On the generation of metabolic novelties in evolution. In: Biogenesis, Evolution, Homeostasis. Alfred Locker, ed. Pergamon Press, New York.

(1974) Biological systems as organizational paradigms. *International Journal of General Systems* 1:165-174

(1978) Fundamentals of Measurement and Representation of Natural Systems. North Holland, New York.

(1981) Pattern generation in networks. *Progress in Theoretical Biology* 6: 161-209.

(1985a) Anticipatory Systems. Pergamon Press, New York.

(1985b) Organisms as causal systems which are not mechanisms: an essay into the nature of complexity. In: Theoretical Biology and Complexity. Three Essays on the Natural Philosophy of Complex Systems. Robert Rosen, ed. Academic Press, Orlando, Florida.

(1986) Causal structures in brains and machines. *International Journal of General Systems* 12: 107-126.

(1987) On the scope of syntactics in mathematics and science: the machine metaphor. In: Real Brains Artificial Minds. J Casti & A Karlqvist, eds. North-Holland, New York.

Rosenblatt, Frank (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review* 65:386-408.

(1962) Strategic approaches to the study of brain models. In: Principles of Self-Organization. Heinz von Foerster, ed. Pergamon Press, Elmsford, NY.

Rucker, Rudy (1982) Infinity and the Mind: The Science and Philosophy of the Infinite. Bantam Books, Toronto.

Rumelhart, DE, G Hinton & JL McClellan (1986) General framework for parallel distributed processing. In:Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol I: Foundations. DE Rumelhart, JL McClellan & PDP Research Group, eds. MIT Press, Cambridge, MA.

Russell, Bertrand (1948) Human Knowledge: Its Scope and Limits. George Allen and Unwin, London.

Schlick, Morris (1925) General Theory of Knowledge. Albert E. Blumberg, trans. Open Court, La Salle, IL, 1985.

Searle, John R. (1980) Minds, brains, and programs. *The Behavioral and Brain Sciences* 3:417-424. Reprinted in: Mind Design. J Haugeland, ed. Bradford Books, Montgomery, VT.

Selfridge, Oliver G (1958) Pandemonium: a paradigm for learning. *Mechanization of Thought Processes: Proc. for a Symposium Held at the National Physical Laboratory, November, 1958,* HMSO, London. Reprinted in: Anderson & Rosenfeld (1988).

(1982) Some themes and primitives in ill-defined systems. In: Adaptive Control of Ill-Defined Systems. Oliver G. Selfridge, Edwina L. Rissland, & MA Arbib, eds. Plenum Press, New York.

Shanker, SG (1987a) Wittgenstein and the Turning Point in the Philosophy of Mathematics. State University of New York Press, Albany, NY.

(1987b) The decline and fall of the mechanist metaphor. In: Artificial Intelligence: The Case Against. Rainer Born, ed. Croom Helm, London.

(1988) Wittgenstein's remarks on the significance of Gödel's Theorem. In: Gödel's Theorem in Focus. SG Shanker, ed. Croom Helm, London.

Shöner, G & JAS Kelso (1988) Dynamic pattern generation in behavioral and neural systems. *Science* 239: 1513-1520 (25 March 1988).

Simon, Herbert A. (1981) The Sciences of the Artificial. Second Edition. MIT Press, Cambridge, MA.

Singh, Jagjit (1966) Great Ideas in Information Theory, Language, and Cybernetics. Dover, New York.

Skarda, Christine A. and Walter J. Freeman (1987) How brains make chaos to make sense of the world. *Behavior and Brain Sciences* 10:161-195.

Smith, Brian (1986) Commentary [on Newell, 1986]: The link from symbols to knowledge. In: Meaning and Cognitive Structure: Issues in the Computational Theory of Mind. Z Pylyshyn & W Demonpoulos, eds. Ablex Publishing, Norwood, NJ.

Swenson, Rod (1989) Emergent attractors and the law of maximum entropy production: foundations of a theory of general evolution. *Systems Research* (in press).

Tamayo & Hartman (1989) Cellular automata, reaction-diffusion systems, and the origin of life. In: Langton, ed. (1989)

Taylor, Michael (1976) Anarchy and Cooperation. John Wiley, New York.

Thagard, Paul (1988) Computational Philosophy of Science. MIT Press, Cambridge, MA.

Thom, Rene (1975) Structural Stability and Morphogenesis: An Outline of a General Theory of Models. Benjamin/Cummings Publishing, Reading, MA.

Thompson, D'Arcy (1917) On Growth and Form. John Tyler Bonner, ed. Cambridge University Press, Cambridge, England, 1966.

Toffoli, Tomaso (1982) Physics and computation. *International Journal of Theoretical Physics* 21(3/4): 165-175.

(1984) Cellular automata as an alternative to (rather than an approximation of) differential equations in modelling physics. *Physica D* 10: 117-127.

Toffoli, Tommaso & Norman Margulis (1987) Cellular Automata Machines. MIT Press, Cambridge, MA.

Tomovic, Rajko (1978) Some control conditions for self-organization--what the control theorist can learn from biology. *American J. Physiology* 235(5): R205-R209.

Tributsch, Helmut (1982) How Life Learned to Live: Adaptation in Nature. MIT Press, Cambridge, MA.

Troelstra, AS and D van Dalen (1988) Constructivism in Mathematics. North-Holland, Amsterdam.

Turing, AM (1936) On the computable numbers with an application to the Entscheidungsproblem. *Proc. London Math. Soc.*, 2nd Ser. 42:230-265. Reprinted in: The Undecidable. Basic Papers. Undecidable Propositions, Unsolvable Problems, and Computable Functions. M Davis, ed. Raven Press, New York, 1965.

(1950) Computing machinery and intelligence. In: Minds and Machines. AR Anderson, ed. Prentice-Hall

Uttley, AM (1958) Conditional probability computing in a nervous system. In: *Mechanization of Thought Processes: Proceedings of a Symposium held at the National Physical Laboratory on 24th, 25th, 26th and 27th of November, 1958*, Her Majesty's Stationary Office, London.

Van Fraassen, Bas C (1980) The Scientific Image. Oxford University Press, UK.

(1985) Empiricism in the philosophy of science. In: Churchland & Hooker, eds. (1985).

Von Helmholtz, Hermann (1878) The facts of perception. In: Helmholtz on Perception: Its Physiology and Development. Richard M Warren and Roslyn P Warren, eds. John Wiley, New York, 1968.

(1894) The origin of the correct interpretation of our sensory impressions. In: Helmholtz on Perception: Its Physiology and Development. Richard M Warren and Roslyn P Warren, eds. John Wiley, New York, 1968.

Von Neumann, John (1931) The formalist foundations of mathematics. In:Philosophy of Mathematics, Selected Readings. 2nd Edition. P Benacerraf & H Putnam, eds. Cambridge University Press, Cambridge, UK, 1983.

(1948) The general and logical theory of automata. In: Papers of John von Neumann on Computing and Computer Theory W Aspray &A Burks, eds. MIT Press, Cambridge, MA, 1987.

(1949) Theory and organization of complicated automata. In: Papers of John von Neumann on Computing and Computer Theory W Aspray & A Burks, eds. MIT Press, Cambridge, MA, 1987.

(1955) Mathematical Foundations of Quantum Mechanics. Princeton University Press, Princeton, NJ. (see Chapter V, General Considerations 1. Measurement and Reversibility pp. 347-358 and Chapter VI, The Measuring Process 1. Formulation of the Problem.)

(1958) The Computer and the Brain Yale University Press, New Haven.

Waddington, CH (1959) Evolutionary adaptation. In: The Evolution of an Evolutionist. Cornell University Press, Ithaca, NY, 1975.

(1972) Postscript. In: Towards a Theoretical Biology. 4. Essays. CH Waddington, ed. Edinburgh University Press, Endinburgh, Scotland.

Weber, Bruce, David Depew & James D Smith, eds. (1988) Entropy, Information, and Evolution: New Perspectives on Physical and Biological Evolution. MIT Press, Cambridge, MA.

Weinand, Richard G & Michael Conrad (1987) Affinity maturation and ecological succession. In: Part One. Theoretical Immunology. Volume II of The Santa Fe Institute for the Sciences of Complexity. *The Proceedings of the Theoretical*

Immunology Workshop held June, 1987 in Santa Fe, New Mexico. AL Perelson, ed. Addison-Wesley, Redwood City, California.

Weinberg, Gerald (1975) Introduction to Systems Thinking. John Wiley, New York.

Wheeler, John Archibald and Wojciech Hubert Zurek (1983) Quantum Theory and Measurement. JA Wheeler & WH Zurek, eds. Princeton University Press, Princeton, NJ.

Wicken, Jeffrey S (1988) Thermodynamics, evolution, and emergence: ingredients for a new synthesis. In Weber, Depew & Smith (1988).

Wilson, Stewart (1986) Knowledge growth in an artificial animal. In: Adaptive and Learning Systems. KS Narendra, ed., Plenum Publishing.

(1987) Classifier systems and the animat problem. *Machine Learning* 2, 1-32.

Wittgenstein, Ludwig (1956) Remarks on the Foundations of Mathematics. GH von Wright, R Rhees, GEM Anscombe, eds. MIT Press, Cambridge, MA, revised edition, 1978.

Wright, Crispin (1980) Wittgenstein on the Foundations of Mathematics. Harvard University Press, Cambridge, MA.

Wysocki, Lawrence, Tim Manser & Malcolm L Gefter (1986) Somatic evolution of variable region structures during an immune response. *PNAS USA* 83: 1847-1851.

Yates, F Eugene (1978) Thermodynamics and life. *American J. Physiol.* 3: R81-R83.

Zeeman, EC (1972) A catastrophe machine. In: Towards a Theoretical Biology. 4. Essays. CH Waddington, ed. Edinburgh University Press, Endinburgh, Scotland. Also in: Catastrophe Theory: Selected Papers 1972-1977. EC Zeeman, Addison Wesley, Reading, MA, 1977.