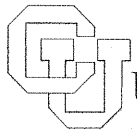
On the Proper Treatment of Connectionism

Paul Smolensky

CU-CS-359-87



University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

Short Abstract

A version of the connectionist approach to cognitive modeling is formulated as a set of hypotheses that confront a number of in-principle attacks and challenges. Attacks confronted include the charges that connectionist models are concerned merely with implementation details, that they are invalid because they are not neurally faithful, that connectionist computation cannot possibly offer anything new, and that connectionist models offer an inadequate kind of explanation. Challenges undertaken include the roles in a connectionist framework for rules, productions, logic, rationality, constituents of mental states, and conceptual schemata or frames.

On the Proper Treatment of Connectionism

Paul Smolensky

CU-CS-359-87 February, 1987

Department of Computer Science &
Institute of Cognitive Science
University of Colorado
Boulder, CO 80309–0430
(303) 492-8991
smolensky@boulder.csnet

Long Abstract

A set of hypotheses is formulated for a connectionist approach to cognitive modeling. These hypotheses are shown to be incompatible with the hypotheses embodied in traditional cognitive models. The connectionist models considered are massively parallel numerical computational systems that are a kind of continuous dynamical system. The numerical variables in the system correspond semantically to fine-grained features below the level of the concepts consciously used to describe the task domain. The level of analysis is intermediate between those of symbolic cognitive models and neural models. The explanations of behavior provided are like those traditional in the physical sciences, unlike the explanations provided by symbolic models. Characterization of cognitive connectionist systems involves a kind of rationality with new consequences for the semantics of mental states. Mental states are seen to possess a new kind of constituent structure.

Higher-level analyses of these connectionist models reveal subtle relations to symbolic models. Fundamentally parallel connectionist memory and linguistic processes are hypothesized to give rise to processes that are describable at a higher level as sequential rule application. At the lower level, computation has the character of massively parallel satisfaction of soft numerical constraints; at the higher level, this can lead to competence characterizable by hard rules. Performance will typically deviate from this competence since behavior is achieved not by interpreting hard rules but by satisfying soft constraints. The result is a picture in which traditional and connectionist theoretical constructs collaborate intimately to provide an understanding of cognition.

Submitted to *The Behavioral and Brain Sciences*

In this paper I formulate a connectionist approach to cognitive modeling—call it *PTC* for the moment. I do not argue the scientific merit of PTC; that some version of connectionism along the lines of PTC constitutes a Proper Description of Processing is extensively argued in the recently published books, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (Rumelhart, McClelland & the PDP Research Group 1986; McClelland, Rumelhart & the PDP Research Group 1986). PTC is instead offered as a Proper Treatment of Connectionism: a coherent formulation of the connectionist approach that puts it in contact with other theory in cognitive science in a particularly constructive way. PTC is intended as a formulation of connectionism that is at once strong enough to constitute a major cognitive hypothesis, comprehensive enough to face a number of difficult challenges, and sound enough to resist a number of objections in principle. If PTC succeeds in these goals, it will facilitate the real business at hand: assessing the scientific adequacy of PTC—determining whether PTC offers adequate computational power for modeling human cognitive competence and appropriate computational mechanisms for accurately modeling human cognitive performance.

PTC is a response to a number of positions that are being adopted concerning connectionism—pro, con, and blandly ecumenical. These positions, which are frequently expressed orally but rarely set down in print, represent, I believe, failures of supporters and critics of the traditional approach truly to come to grips with each other. Advocates of the traditional approach are often willing to grant that connectionist systems are useful, perhaps even important, for modeling lower-level processes (eg. early vision), or for fast and fault-tolerant implementation of conventional AI programs, or for understanding how the brain happens to implement Lisp. I believe that these ecumenical positions fail to recognize the true force of the connectionist assault on the received view of cognition, and PTC is an explicit formulation of this assault. Other supporters of the traditional approach find the connectionist approach to be fatally flawed because it cannot offer anything new (since Universal Turing machines are, after all, Universal), or because it cannot offer the kinds of explanations that cognitive science requires; some dismiss the connectionist approach on the grounds that it is too unfaithful neurally. PTC has been designed to withstand these attacks. On the opposite side, most existing connectionist models fail to come to grips with the traditional approach (partly through a neglect intended as benign). It is easy to read into the connectionist literature the claim that there is no role in cognitive science for traditional theoretical constructs such as rules, sequential processing, logic, rationality, and conceptual schemata or frames. PTC undertakes to assign these constructs their proper role in a connectionist paradigm for cognitive modeling. PTC also attempts to address a number of foundational issues concerning mental states.

I see no way of achieving the goals of PTC without adopting certain positions that will be regarded by a number of connectionists as premature or mistaken. These are inevitable consequences of the fact that the connectionist approach is still quite underdeveloped, and that indeed the term "connectionist" has come to label a number of approaches that embody significantly conflicting assumptions.

Most of the foundational issues surrounding the connectionist approach turn, in one way or another, on the *level of analysis* adopted. The terminology, graphics, and discussion found in most connectionist papers strongly suggest that connectionist modeling operates at the neural level. I will argue, however, that the principles of cognition being explored in the connectionist approach are better *not* construed as the principles of the neural level. Specification of the level of cognitive analysis adopted by PTC is a subtle matter which consumes much of this paper. To be sure, the level of analysis adopted by PTC is lower than that of the traditional, symbolic paradigm; but, at least for the present, the level of PTC is more explicitly related to the level of the symbolic paradigm than it is to the neural level. For this reason I will call the paradigm for cognitive modeling proposed by PTC the *subsymbolic paradigm.*

A few comments on terminology are appropriate. I will refer to the traditional approach to cognitive modeling as the *symbolic paradigm*, intending to emphasize that in this approach, cognitive descriptions are built of entities that are *symbols* both in the semantic sense of referring to external objects and in the syntactic sense of being operated upon by "symbol manipulation." The name "subsymbolic paradigm" is intended to suggest cognitive descriptions built up of *constituents* of the symbols used in the symbolic paradigm; these fine-grained constituents might be called *subsymbols*. Entities that are typically represented in the symbolic paradigm by symbols are typically represented in the subsymbolic paradigm by a large number of subsymbols. Along with this semantic distinction comes a syntactic distinction. Subsymbols are not operated upon by "symbol manipulation": they participate in numerical—not symbolic—computation. Operations that in the symbolic paradigm consist of a single discrete operation (eg. a memory fetch) are often achieved in the subsymbolic paradigm as the result of a large number of much finer-grained (numerical) operations.

While the level of cognitive analysis adopted by the subsymbolic paradigm for formulating connectionist models is lower than the level traditionally adopted by the symbolic paradigm, for the purposes of relating these two paradigms it is often important to analyze connectionist models at a higher level; to amalgamate, so to speak, the subsymbols into symbols. While the symbolic and subsymbolic paradigms each have their preferred level of

analysis, the cognitive models they offer can each be described at multiple levels. It is therefore useful to have distinct names for the levels: I will call the preferred level of the symbolic paradigm the *conceptual level* and that of the subsymbolic paradigm the *subconceptual level*. These names are not ideal, but will be further motivated in the course of characterizing these levels. A primary goal of this paper is to articulate a coherent set of hypotheses about the subconceptual level: the kind of cognitive descriptions that are used, the computational principles that apply, and the relations between the subconceptual and both the symbolic and neural levels.[1]

The choice of level greatly constrains the appropriate formalism for analysis. Probably the most striking feature of the connectionist approach is the change in formalism relative to the symbolic paradigm. Since the birth of cognitive science, *language* has provided the dominant theoretical model. Formal cognitive models have taken their structure from the syntax of formal languages, and their content from the semantics of natural language. The mind has been taken to be a machine for formal symbol manipulation, and the symbols manipulated have assumed essentially the same semantics as words of English.

The subsymbolic paradigm challenges both the syntactic and semantic role of language in formal cognitive models. Section 1 formulates this challenge. Alternative fillers are described for the roles language has traditionally played in cognitive science, and the new role left to language is delimited. The fundamental hypotheses defining the subsymbolic paradigm are formulated, and the challenge that nothing new is being offered is considered. Section 2 considers the relation between the subsymbolic paradigm and neuroscience; the challenge that connectionist models are too neurally unfaithful is addressed. Section 3 presents the relations between analyses of cognition at the neural, subconceptual, and conceptual levels. The remainder of the paper, which deals with the relations between the subconceptual and conceptual levels, is previewed, and the types of explanations of behavior provided by the symbolic and subsymbolic paradigms are discussed. Section 4 faces the challenge of accounting for conscious, rule-guided behavior within the subsymbolic paradigm. Section 5 addresses the challenge of distinguishing cognitive from non-cognitive systems at the subconceptual level. Various properties of subsymbolic mental states, and the issue of rationality, are considered. Section 6 elaborates briefly on the computational principles that apply at the subconceptual level. Section 7 discusses how higher, conceptual-level descriptions of subsymbolic models approximate symbolic models (under their conceptual-level descriptions).

In this paper I have tried to typographically isolate concise formulations of the main points. Most of these numbered points serve to characterize the subsymbolic paradigm, but a few define opposing points of view; to avoid confusion, the latter have been explicitly tagged: *to be rejected.*

1. Formalization of knowledge

1.1. Cultural knowledge and conscious rule interpretation

What is an appropriate formalization of the knowledge cognitive agents possess and the means by which they use that knowledge to perform cognitive tasks? As a starting point, we can look to those knowledge formalizations that predate cognitive science. The most formalized knowledge is found in sciences like physics that rest on mathematical principles. Domain knowledge is formalized in linguistic structures like "energy is conserved" (or an appropriate encryption), and logic formalizes the use of that knowledge to draw conclusions. Knowledge consists of axioms, and drawing conclusions consists of proving theorems.

This method of formulating knowledge and drawing conclusions has extremely valuable properties:

(1) a. *Public access:* The knowledge is accessible to many people.
b. *Reliability:* Different people (or the same person at different times) can reliably check whether conclusions have been validly reached.
c. *Formality; bootstrapping, universality:* The inference operations require very little experience with the domain to which the symbols refer.

These three properties are important for science because it is a *cultural activity.* It is of limited social value to have knowledge that resides purely in one individual (1a). It is of questionable social value to have knowledge formulated in such a way that different users draw different conclusions (eg., can't agree that an experiment falsifies

a theory) (1b). For cultural propagation of knowledge, it is helpful if novices with little or no experience with a task can be given a means for performing that task, and thereby a means for acquiring experience (1c).

There are other cultural activities besides science with similar requirements. The laws of a nation and the rules of an organization are also linguistically formalized procedures for effecting action which different people can carry out with reasonable reliability. In all these cases, the goal is to create an abstract decision system that resides outside any single person.

Thus *at the cultural level*, the goal is to express knowledge in a form that can be executed reliably by different people, even inexperienced ones. We can view the top-level conscious processor of individual people as a *virtual machine*—the *conscious rule interpreter*—and we can view cultural knowledge as a program that runs on that machine. Linguistic formulations of knowledge are perfect for this purpose. The procedures different people can reliably execute are explicit, step-by-step linguistic instructions. This is what has been formalized in the theory of *effective procedures*. Thanks to property (1c), the top-level conscious human processor can be idealized as *universal*: capable of executing any effective procedure. The theory of effective procedures—the classical theory of computation—is physically manifest in the von Neumann computer. One can say that the von Neumann computer is a machine for automatically following the kind of explicit instructions that people can fairly reliably follow—but much faster and with perfect reliability.

Thus we can understand why the production system of computation theory, or more generally the von Neumann computer, has provided a successful model of how people execute instructions (eg., models of novice physics problem solving such as Larkin, McDermott, Simon & Simon 1980). In short, when people (eg. novices) consciously and sequentially follow rules (eg. that they have been taught), their cognitive processing is naturally modeled as the sequential interpretation[2] of a linguistically formalized procedure. The rules being followed are expressed in terms of the consciously accessible concepts with which the task domain is conceptualized. In this sense, the rules are formulated at the *conceptual level* of analysis.

To sum up:

(2) a. Rules formulated in natural language can provide an effective formalization of cultural knowledge.
b. Conscious rule application can be modeled as the sequential interpretation of such rules by a virtual machine called the conscious rule interpreter.
c. These rules are formulated in terms of the concepts consciously used to describe the task domain—they are formulated at the conceptual level.

1.2. Individual knowledge, skill, and intuition in the symbolic paradigm

But the constraints on *cultural knowledge formalization* are not the same as those on *individual knowledge formalization*. The intuitive knowledge in a physics expert or a native speaker may demand, for a truly accurate description, a formalism that is not a good one for cultural purposes. After all, the individual knowledge in an expert's head does not possess the properties (1) of cultural knowledge: it is not publically accessible, is not completely reliable, and is completely dependent on ample experience. Individual knowledge is a program that runs on a virtual machine that need not be the same as the top-level conscious processor that runs the cultural knowledge. By definition, conclusions reached by intuition do not come from conscious application of rules, and intuitive processing need not have the same character as conscious rule application.

What kinds of programs are responsible for behavior that is not conscious rule application? I will refer to the virtual machine that runs these programs as the *intuitive processor*. It is (presumably) responsible for all of animal behavior, and a huge portion of human behavior: perception, practiced motor behavior, fluent linguistic behavior, intuition in problem solving and game playing—in short, practically all of skilled performance. The transference of responsibility from the conscious rule interpreter to the intuitive processor during the acquisition of skill is one of the most striking and well-studied phenomena in cognitive science (eg., Anderson 1981). An analysis of the formalization of knowledge must consider both the knowledge involved in novices' conscious application of rules and the knowledge resident in experts' intuition, as well as their relationship.

An appealing possibility is this:

(3) a. The programs running on the intuitive processor consist of linguistically formalized rules that are sequentially interpreted. *(To be rejected.)*

This has traditionally been the assumption of cognitive science. Native speakers are unconsciously interpreting rules, as are physics experts when they are intuiting answers to problems. Artificial intelligence systems for natural language processing and problem solving are programs written in a formal language for the interpretation of symbolic descriptions of procedures for manipulating symbols.

To the syntactic hypothesis (3a) there corresponds a semantic one:

(3) b. The programs running on the intuitive processor are composed of elements—symbols—referring to essentially the same concepts as are used to consciously conceptualize the task domain. *(To be rejected.)*

This applies to production system models in which the productions representing expert knowledge are compiled versions of those of the novice (eg., Anderson 1983; Lewis 1978) and to the bulk of AI programs.

Hypotheses (3a) and (3b) together comprise

(3) **The unconscious rule interpretation hypothesis:** *(To be rejected.)*
The programs running on the intuitive processor have a syntax and semantics comparable to those running on the conscious rule interpreter.

This hypothesis has provided the foundation for the symbolic paradigm for cognitive modeling. Cognitive models of both conscious rule application and intuitive processing have been programs constructed of entities which are *symbols* both in the syntactic sense of being operated on by "symbol manipulation" and in the semantic sense of (3b). Because these symbols have the conceptual semantics of (3b), I will call the level of analysis at which these programs provide cognitive models the *conceptual level.*

1.3. The subsymbolic paradigm and intuition

The hypothesis of unconscious rule interpretation (3) is an attractive possibility which a connectionist approach to cognitive modeling rejects. Since my purpose here is to formulate rather than argue the scientific merits of a connectionist approach, I will not argue against (3) here. I will point out only that in general, connectionists do not casually reject (3). Several of today's leading connectionist researchers were intimately involved with serious and longstanding attempts to make (3) serve the needs of cognitive science.[3] Connectionists tend to reject (3) because they find the consequences that have actually resulted from its acceptance to be quite unsatisfactory, for a number of quite independent reasons, for example:

(4) a. Actual AI systems built on hypothesis (3) seem too brittle, too inflexible, to model true human expertise.
b. The process of articulating expert knowledge in rules seems impractical for many important domains (eg., common sense).
c. Hypothesis (3) has contributed essentially no insight into how knowledge is represented in the brain.

What motivates pursuit of connectionist alternatives to (3) are hunches that such alternatives will better serve the goals of cognitive science. Comprehensive empirical assessment of these hunches are probably at least a decade away.

One possible alternative to (3a) is

(5) **The neural architecture hypothesis:** *(To be rejected.)*
The intuitive processor for a particular task uses the same architecture that the brain employs for that task.

Whatever appeal this hypothesis might have, it seems incapable in practice of supporting the needs of the vast majority of cognitive models. We simply do not know what architecture the brain uses for performing most cognitive tasks. There may be some exceptions (like vision and spatial tasks), but for problem solving, language, and many others (5) simply cannot now do the necessary work.

These points and others relating to the neural level will be considered in more detail in section 2. For now the point is simply that viably characterizing the level of analysis of connectionist modeling is not trivially a matter of identifying it with the neural level. While the level of analysis adopted by most connectionist cognitive models is not the conceptual level, it is also not the neural level.

The goal now is to formulate a connectionist alternative to (3) that, unlike (5), provides a viable basis for cognitive modeling. A first, crude cut at this hypothesis is:

(6) The intuitive processor possesses a certain kind of connectionist architecture (which abstractly models a few of the most general features of neural networks). *(To be elaborated.)*

Postponing the parenthetical remark to section 2, we now consider the relevant kind of connectionist architecture.

The kind of connectionist model I will consider can be described as a network of very simple processors, *units*, each possessing a numerical *activation value* that is dynamically determined by the values of the other processors in the network. The *activation equation* governing this interaction has numerical parameters which determine the direction and magnitude of the influence of one activation value on another; these parameters are called the *connection strengths* or *weights*. The activation equation is a differential equation (usually approximated by the finite difference equation that arises from discrete time slices; the issue of discrete approximation is taken up in Section 6.1). The weights modulate the behavior of the network: they constitute the "program" for this architecture. A network is sometimes programmed by the modeler, but often a network programs itself to perform a task by changing its weights in response to examples of input/output pairs for the task. The *learning rule* is the differential equation governing the weight changes.

The knowledge in a connectionist system lies in its connection strengths. Thus for the first part of our elaboration on (6) we have the following alternative to (3a):

(7) a. **The connectionist dynamical system hypothesis:**
The state of the intuitive processor at any moment is precisely defined by a vector of numerical values (one for each unit). The dynamics of the intuitive processor is governed by a differential equation. The numerical parameters in this equation constitute the processor's program or knowledge. These parameters may change according to a learning equation.

This hypothesis states that the intuitive processor is a certain kind of dynamical system, with the same general character as dynamical systems traditionally studied in physics. The special properties that distinguish this kind of dynamical system—a *connectionist dynamical system*—are only vaguely described in (7a), and a more precise specification is needed. It is premature at this point to commit to such a specification, but one large class of subsymbolic models is that of *quasi-linear dynamical systems*, explicitly discussed in Smolensky (1986b) and Rumelhart, Hinton and Williams (1986). Each unit in a quasi-linear system computes its value by first calculating the weighted sum of its inputs from other units, and then transforming this sum with a non-linear function. An important goal is to characterize the computational properties of various kinds of connectionist dynamical systems (such as quasi-linear systems) and to thereby determine which kinds provide models of various types of cognitive processes.

The connectionist dynamical system hypothesis (7a) provides a connectionist alternative to the syntactic hypothesis (3a) of the symbolic paradigm. We now need a semantic hypothesis compatible with (7a) to replace (3b). The question is: What does a unit's value *mean?* The most straightforward possibility is that the semantics of

each unit is comparable to that of a natural language word; each unit represents such a concept, and the connection strengths between units reflect the "degree of association" between the concepts.

(8) **The conceptual unit hypothesis:** *(To be rejected.)*
Individual intuitive processor elements—individual units—have essentially the same semantics as the conscious rule interpreter's elements—words of natural language.

But (7a) and (8) make an infertile couple. Activation of concepts spreading along "degree of association" links may be adequate for modeling simple aspects of cognition—like relative times for naming words in various contexts, or the relative probabilities of perceiving letters in various contexts—but it cannot be adequate for complex tasks like question answering or grammaticality judgements. The relevant structures cannot even be feasibly represented in such a network, let alone effectively processed.

Great computational power must be present in the intuitive processor to deal with the many cognitive processes that are extremely complex when described at the conceptual level. The symbolic paradigm, based on hypothesis (3), gets its power by allowing highly complex, essentially arbitrary, operations on symbols with conceptual-level semantics: simple semantics, complex operations. If the operations are required to be as simple as those allowed by hypothesis (7a), we cannot get away with a semantics as simple as that of (8).[4] A semantics compatible with (7a) must be more complicated:

(7) b. **The subconceptual unit hypothesis:**
The entities in the intuitive processor with the semantics of conscious concepts of the task domain are *complex patterns of activity over many units*. Each unit participates in many such patterns.

(See Hinton, McClelland & Rumelhart 1986, and several of the papers in Hinton & Anderson 1981; the neural counterpart is associated with Hebb 1949 and Lashley 1950, about which see Feldman 1986.) The interactions between *individual units* are simple, but these units do not have conceptual semantics: they are *subconceptual*. The interactions between the entities with conceptual semantics—interactions between complex patterns of activity—are not at all simple. Interactions at the level of activity patterns are not directly described by the formal definition of a subsymbolic model; they must be computed by the analyst. Typically, these interactions can be computed only approximately. There will generally be no precisely valid, computable formal principles at the conceptual level; such principles exist only at the level of the units—the *subconceptual level*.

(7) c. **The subconceptual level hypothesis:**
Precise, formal descriptions of the intuitive processor are generally tractable not at the conceptual level, but only at the subconceptual level.

Hypotheses (7a-c) can be summarized as

(7) **The subsymbolic hypothesis:**
The intuitive processor is a subconceptual connectionist dynamical system that does not admit a precise formal conceptual-level description.

This hypothesis is the cornerstone of the subsymbolic paradigm.[5]

1.4. The incompatibility of the symbolic and subsymbolic paradigms

I will now show that the symbolic and subsymbolic paradigms, as formulated above, are incompatible—the hypotheses (3) and (7) about the syntax and semantics of the intuitive processor are not mutually consistent. This issue requires care, since it is well known that one virtual machine can often be implemented in another, that a program written for one machine can be translated into a program for the other. The attempt to distinguish subsymbolic and symbolic computation might well be futile if each can simulate the other. After all, a digital computer is in reality some sort of dynamical system simulating a von Neumann automaton, and in turn digital computers are usually used to simulate connectionist models. Thus it seems possible that the symbolic and subsymbolic hypotheses (3) and (7) are *both* correct: that the intuitive processor can be regarded as a virtual machine for sequentially interpreting rules on one level *and* as a connectionist machine on a lower level.

This possibility fits comfortably within the symbolic paradigm, under a formulation such as

(9) Valid connectionist models are merely implementations, for a certain kind of parallel hardware, of symbolic programs that provide exact and complete accounts of behavior at the conceptual level. *(To be rejected.)*

However (9) contradicts hypothesis (7c), and is thus fundamentally incompatible with the subsymbolic paradigm. The symbolic programs that (3) hypothesizes for the intuitive processor could indeed be translated for a connectionist machine, but the translated programs would *not* be the kind of subsymbolic program that (7) hypothesizes.

What about the reverse relationship, where a symbolic program is used to implement a subsymbolic system? Here it is crucial to realize that the symbols in such programs represent the activation values of units and the strengths of connections. By hypothesis (7b), these do not have conceptual semantics, and thus hypothesis (3b) is violated. The subsymbolic programs that (7) hypothesizes for the intuitive processor can be translated for a von Neumann machine, but the translated programs are *not* the kind of symbolic program that (3) hypothesizes.

These arguments show that unless the hypotheses of the symbolic and subsymbolic paradigms are formulated with some care, the substance of the scientific issue at stake can easily be missed. It is well known that von Neumann machines and connectionist networks can simulate each other. If one cavalierly characterizes the approaches *only syntactically*, using (3a) and (7a) alone, then indeed the issue—connectionist or not connectionist—appears to be "one of AI's wonderful red herrings."[6] It is a mistake to claim that the connectionist approach has nothing new to offer cognitive science. The issue at stake is a central one: *Do the formal principles of cognition lie at the conceptual level?* The answer offered by the subsymbolic paradigm is: *No—they lie at the* **sub***conceptual level.*

2. The subconceptual and neural levels

Hypothesis (7b) leaves open important questions about the semantics of subsymbolic systems. What kind of subconceptual features do the units in the intuitive processor represent? Which activity patterns actually correspond to particular concepts?

Each individual subsymbolic model has adopted particular procedures for relating patterns of activity—activity vectors—to the conceptual-level descriptions of inputs and outputs that define the model's task. The vectors chosen are often vectors of values of fine-grained features of the inputs and outputs, based on some pre-existing theoretical analysis of the domain. For example, for the task studied in Rumelhart and McClelland (1986), transforming root phonetic forms of English verbs to their past-tense forms, the input and output phonetic strings were represented as vectors of values for context-dependent binary phonetic features. The task description at the conceptual level involves consciously available concepts such as the words "go" and "went," while the subconceptual level employed by the model involved a very large number of fine-grained features such as "roundedness preceded by frontalness and followed by backness." The representation of "go" is a large pattern of activity over these features.

Substantive progress in subsymbolic cognitive science requires that systematic commitments to vectorial representations be made for individual cognitive domains. The vectors chosen to represent inputs and outputs crucially affect a model's predictions, since the generalizations the model makes are largely determined by the similarity structure of the chosen vectors. Unlike symbolic tokens, these vectors lie in a topological space, in which some are close together and others far apart.

It might seem that the mapping between patterns of activity and conceptual-level interpretations ought to be determined by neuroscience. This brings us back to the parenthetical comment in (6) and the general issue of the relation between the subconceptual and neural levels.

To what extent does the architecture of the intuitive processor assumed in the subsymbolic paradigm (7a) match the structure of the brain? Table 1 presents some of the relations. The left column lists currently plausible features of some of the most general aspects of the neural architecture, considered at the level of neurons (see, eg., Crick 1986). The right column lists the corresponding architectural features of the connectionist dynamical systems

typically employed in subsymbolic models. In the center column, hits have been indicated by + and misses by –.

Insert Table 1 about here

In Table 1 the loose correspondence assumed is between neurons and units, between synapses and connections. It is not clear how to make this correspondence precise. Does the activity of a unit correspond to the membrane potential at the cell body? Or the time-averaged firing rate of the neuron? Or the population-averaged firing rate of many neurons? Since the integration of signals between dendritic trees is probably more like the linear integration appearing in quasi-linear dynamical systems than is the integration of synaptic signals on a dendrite, wouldn't it be better to view a connection not as an individual synaptic contact but rather as an aggregate contact on an entire dendritic tree?

Given the difficulty of precisely stating the neural counterpart of components of subsymbolic models, and given the significant number of misses even in the very general properties considered in Table 1, it seems advisable to keep open the question of the detailed relation between cognitive descriptions at the subconceptual and neural levels. There seems no denying, however, that the subconceptual level is significantly closer to the neural level than is the conceptual level; symbolic models possess even fewer similarities with the brain than those indicated in Table 1.

The subconceptual level ignores a great number of features of the neural level that are probably extremely important to understanding how the brain computes. Nonetheless, the subconceptual level does incorporate a number of features of neural computation that are almost certainly extremely important to understanding how the brain computes. The general principles of computation at the subconceptual level—computation in high-dimensional, high-complexity dynamical systems—*must* apply to computation in the brain; these principles are likely to be necessary, even if not sufficient, to understanding neural computation. And while subconceptual principles are not unambiguously and immediately applicable to neural systems, they are certainly more readily applicable than the principles of symbolic computation. In sum:

(10) The fundamental level of the subsymbolic paradigm, the subconceptual level, lies between the neural and conceptual levels.

It is common to hear dismissals of a particular subsymbolic model because it is not immediately apparent how to precisely implement it in neural hardware, or because certain neural features are absent from the model. We can now identify two fallacies in such a dismissal. First, following (10): *Subsymbolic models should not be viewed as neural models.* If the subsymbolic paradigm proves valid, the best subsymbolic models of a cognitive process should one day be shown to be some reasonable higher-level approximation to the neural system supporting that process. This provides a heuristic that favors subsymbolic models that seem more likely to be reducible to the neural level. But this heuristic is an extremely weak one given how difficult such a judgement must be with the current confusion about the precise neural correlates of units and connections, and the current state of both empirical and theoretical neuroscience.

The second fallacy rests on a failure to recognize the role of individual models in the subsymbolic paradigm. An extremely significant contribution of a model is providing evidence for general principles that are characteristic of a broad class of subsymbolic systems. The potential value of "ablation studies" of the NETtalk text-to-speech system (Sejnowski & Rosenberg 1986) does not depend entirely on the neural faithfulness of the model, or even on its psychological faithfulness. NETtalk is a subsymbolic system that performs a very complex task. What happens to its performance when parts of its innards are damaged? This provides an important clue to the general principles of degradation in *all* complex subsymbolic systems: principles that will apply to future systems that are more faithful as models.

It should be pointed out that there are, of course, many neural models that *do* take seriously many of the constraints of neural organization, and for which the analogue of Table 1 would show nearly all hits. But we are concerned here with connectionist models for performing cognitive tasks, and these models typically possess the features displayed in the table. The claim is not that neural models don't exist, but rather that they should not be confused with subsymbolic models.

Why is it that neural models of cognitive processes are, generally speaking, currently infeasible? The problem is not an insufficient quantity of data about the brain. The problem, it seems, is that this data is generally of the *wrong kind* for cognitive modeling. Our information about the nervous system tends to describe its *structure*, not its *dynamic behavior*. Subsymbolic systems are dynamical systems with certain kinds of differential equations governing their dynamics. If we knew what dynamical variables in the neural system for some cognitive task were the critical ones for performing that task, and what the "equations of motion" were for those variables, we could use that information to build neurally faithful cognitive models. But generally what we know instead are endless static properties of how the hardware is arranged. Without knowing which (if any) of these structures support relevant dynamical processes, and what equations govern these processes, one suspects we are in a position comparable to someone attempting to model the solar system, armed with voluminous data on the colored bands of the planets but with no knowledge of Newton's Laws.

To summarize:

(11) a. Unlike the symbolic architecture, the subsymbolic architecture possesses a number of the most general features of the neural architecture.
b. However, the subsymbolic architecture lacks a number of the more detailed but still quite general features of the neural architecture; the subconceptual level of analysis is higher than the neural level.
c. For most cognitive functions, neuroscience cannot now provide the relevant information to specify a cognitive model at the neural level.
d. The general cognitive principles of the subconceptual level will likely be important contributors to future discoveries of those specifications of neural computations that we now lack.

3. Reduction of cognition to the subconceptual level

The previous section considered the relation between the fundamental level of the subsymbolic paradigm—the subconceptual level—and the neural level. The remainder of the paper will focus on relations between the subconceptual and conceptual levels, which has so far only been briefly touched upon (in (7c)). Before proceeding, however, it is worth concisely summarizing the relationships between the levels, including those that will be discussed in the remainder of the paper.

Imagine three physical systems: a brain that is executing some cognitive process, a massively parallel connectionist computer running a subsymbolic model of that process, and a von Neumann computer running a symbolic model of the same process. The cognitive process may involve conscious rule application, intuition, or a combination of the two. According to the subsymbolic paradigm, here are the relationships:

(12) a. Describing the brain at the neural level gives a neural model.
b. Describing the brain approximately, at a higher level—the subconceptual level—yields, to a good approximation, the model running on the connectionist computer, when it too is described at the subconceptual level. (At this point, this is a goal for future research. It could turn out that the degree of approximation here is only rough; this would still be consistent with the subsymbolic paradigm.)
c. We can try to describe the connectionist computer at a higher level—the conceptual level—by using the patterns of activity that have conceptual semantics. If the cognitive process being executed is conscious rule application, we will be able to carry out this conceptual level analysis with reasonable precision, and will end up with a description that closely matches the symbolic computer program running on the von Neumann machine.

d. If the process being executed is an intuitive process, we will be unable to carry out the conceptual-level description of the connectionist machine precisely. Nonetheless, we will be able to produce various approximate conceptual-level descriptions that correspond in various ways to the symbolic computer program running on the von Neumann machine.

For a cognitive process involving both intuition and conscious rule application, (12c) and (12d) will each apply to certain aspects of the process.

The relationships (12a) and (12b) were discussed in the previous section. The relationship (12c) between a subsymbolic implementation of the conscious rule interpreter and a symbolic implementation are discussed in Section 4. The relations (12d) between subsymbolic and symbolic accounts of intuitive processing are considered in Section 7. These relations hinge on certain subsymbolic computational principles operative at the subconceptual level (12b); these are briefly discussed in Section 6. These principles are of a new kind for cognitive science, giving rise to the foundational considerations taken up in Section 5.

The relationships in (12) can be more clearly understood by reintroducing the concept of "virtual machine." If we take one of the three physical systems and describe its processing at a certain level of analysis, we get a virtual machine that I will denote "system$_{level}$". Then (12) can be written:

(13) a. brain$_{neural}$ = neural model
b. brain$_{subconceptual}$ ≈ connectionist$_{subconceptual}$
c. connectionist$_{conceptual}$ ≈ von Neumann$_{conceptual}$ (conscious rule application)
d. connectionist$_{conceptual}$ ~ von Neumann$_{conceptual}$ (intuition)

Here, the symbol "≈" means "equals to a good approximation" and "~" means "equals to a crude approximation." The two nearly equal virtual machines in (13c) both describe what we have been calling the "conscious rule interpreter." The two roughly similar virtual machines in (13d) provide the two paradigms' descriptions of the intuitive processor at the conceptual level.

Table 2 indicates these relationships and also the degree of exactness to which each system can be described at each level—the degree of precision to which each virtual machine is defined. The levels included in Table 2 are those relevant to predicting high-level behavior. Of course each system can also be described at lower levels, all the way down to elementary particles. However, levels below an exactly describable level are ignorable from the point of view of predicting high-level behavior, since it is possible (in principle) to do the prediction at the highest level that can be exactly described (and it is presumably much harder to do the same at lower levels). This is why in the symbolic paradigm any descriptions below the symbolic level are not viewed as significant. For modeling high-level behavior, how the symbol manipulation happens to be implemented can be ignored—it is not a relevant part of the cognitive model. In a subsymbolic model, exact behavioral prediction must be performed at the subconceptual level—but how the units happen to be implemented is not relevant.

Insert Table 2 about here

The relation between the conceptual level and lower levels is fundamentally different in the subsymbolic and symbolic paradigms. This leads to important differences in the kind of explanations the paradigms offer of conceptual-level behavior, and the kind of reduction used in these explanations. A symbolic model is a *system* of interacting processes, all with the same conceptual-level semantics as the task behavior being explained. Adopting the terminology of Haugeland (1978), this *systematic explanation* relies on a *systematic reduction* of the behavior that involves no shift of semantic domain or *dimension*. Thus a game-playing program is composed of subprograms that generate possible moves, evaluate them, and so on. In the symbolic paradigm, these systematic reductions play the major role in explanation. The lowest-level processes in the systematic reduction, still with the original semantics of the task domain, are then themselves reduced by *intentional instantiation:* they are implemented exactly by other processes with different semantics but the same form. Thus a move-generation subprogram with

game semantics is instantiated in a system of programs with list-manipulating semantics. This intentional instantiation typically plays a minor role in the overall explanation, if indeed it is regarded as a cognitively relevant part of the model at all.

Thus cognitive explanations in the symbolic paradigm rely primarily on reductions involving no dimension shift. This feature is not shared by the subsymbolic paradigm, where accurate explanations of intuitive behavior require descending to the subconceptual level. The elements in this explanation, the units, do *not* have the semantics of the original behavior: that is the content of the subconceptual unit hypothesis, (7b). Thus

(14) Unlike symbolic explanations, subsymbolic explanations rely crucially on a semantic ("dimension") shift that accompanies the shift from the conceptual to the subconceptual levels.

The overall dispositions of cognitive systems are explained in the subsymbolic paradigm as approximate higher level regularities that emerge from quantitative laws operating at a more fundamental level with different semantics. This is the kind of reduction familiar in natural science, exemplified by the explanation of the laws of thermodynamics through a reduction to mechanics that involves shifting dimension from thermal semantics to molecular semantics. (Section 7 discusses some explicit subsymbolic reductions of symbolic explanatory constructs.)

Indeed the subsymbolic paradigm repeals the other features that Haugeland identified as newly introduced into scientific explanation by the symbolic paradigm. The inputs and outputs of the system are not "quasilinguistic representations" but good old-fashioned numerical vectors. These inputs and outputs have semantic interpretations, but these are not constructed recursively from interpretations of imbedded constituents. And the fundamental laws are good old-fashioned numerical equations.

Haugeland went to considerable effort to legitimize the form of explanation and reduction used in the symbolic paradigm. The explanations and reductions of the subsymbolic paradigm, by contrast, are of a type well-established in natural science.

In summary, let me emphasize that in the subsymbolic paradigm the conceptual and subconceptual levels are not related as the levels of a von Neumann computer (high-level-language program, compiled low-level program, etc.). The relationship between subsymbolic and symbolic models is more like that between quantum and classical mechanics. Subsymbolic models accurately describe the microstructure of cognition, while symbolic models provide an approximate description of the macrostructure. An important job of subsymbolic theory is to delineate the situations and respects in which the symbolic approximation is valid, and to explain why.

4. Conscious rule application in the subsymbolic paradigm

In the symbolic paradigm, both conscious rule application and intuition are described at the conceptual level: as conscious and unconscious rule interpretation, respectively. In the subsymbolic paradigm, conscious rule application can be formalized at the conceptual level but intuition must be formalized at the subconceptual level. This suggests that a subsymbolic model of a cognitive process involving both intuition and conscious rule interpretation would consist of two components employing quite different formalisms. While this hybrid formalism might have considerable practical value, there are some theoretical problems with it. How would the two formalisms communicate? How would the hybrid system evolve with experience, reflecting the development of intuition and the subsequent remission of conscious rule application? How would the hybrid system elucidate the fallibility of actual human rule application (eg. logic)? How would the hybrid system get us closer to understanding how conscious rule application is achieved neurally?

All these problems can be addressed by adopting a unified subconceptual-level analysis of both intuition and conscious rule interpretation. The virtual machine that is the conscious rule interpreter is to be implemented in a lower-level virtual machine: the same connectionist dynamical system that models the intuitive processor. How this can, in principle, be achieved is the subject of this section. The relative advantages and disadvantages of implementing the rule interpreter in a connectionist dynamical system rather than a von Neumann machine will also be considered.

The observation is this.

(15) The competence to represent and process linguistic structures in a native language is a competence of the human intuitive processor, so the subsymbolic paradigm assumes that this competence can be modeled in a subconceptual connectionist dynamical system. By combining such linguistic competence with existing memory capabilities of connectionist systems, sequential rule interpretation can be implemented.

Assuming that sentences of natural language can be represented in a subconceptual connectionist dynamical system means that such sentences correspond to certain patterns of activity. Assuming that sentences can be processed means in particular that a pattern of activity representing a verbal instruction can be used to carry out that instruction. Once sentences are represented as patterns of activity, the well-known procedures of associative memories can be used to store them. These are content-addressable memories in which reinstantiation of a part of the stored item causes reinstantiation of the complete item. A collection of such memories can then be used to drive sequential behavior, as follows.

First, a set of linguistically expressed rules is presented to the connectionist system and thereby stored. For concreteness we can imagine the rules to be productions: "if *condition* holds, then do *action*." In a particular situation when the condition of a rule holds, the pattern of activity representing the condition will be instantiated in the network. This will cause the entire pattern of activity representing that rule to be reinstantiated by the memory retrieval mechanism. Now it is as if the rule had been linguistically presented to the system from an external instructor. The language processing mechanism can interpret the sentence, generating the appropriate action and leading to a new pattern of activity in the network representing the new situation. This new pattern leads to the reinstantiation of another stored rule, and the cycle repeats.

Using the stored rules the network can perform the task. The standard learning procedures of connectionist models turn this experience performing the task into a set of weights for going from inputs to outputs. Eventually, after enough experience, the task can be performed directly by these weights. The input activity generates the output activity so quickly that before the relatively slow interpretation process has a chance to reinstantiate the first rule and carry it out, the task is done. With intermediate amounts of experience, some of the weights are well enough in place to prevent some of the rules from having the chance to instantiate, while others are not, enabling other rules to be retrieved and interpreted.

4.1. Rule interpretation, consciousness, and seriality

What about the conscious aspect of rule interpretation? Since consciousness seems to be a quite high-level description of mental activity, it is reasonable to suspect that it reflects the very coarse structure of the cognitive dynamical system. Considering coarseness on the time dimension, we are lead to hypothesize:

(16) Patterns of activity that are stable for relatively long periods of time (on the order of 100 msec) determine the contents of consciousness.

(See Rumelhart, Smolensky, McClelland & Hinton 1986.) The rule interpretation process requires the maintenance of the retrieved linguistically coded rule while it is being carried out. Thus the pattern of activity representing the rule is stable for a relatively long time. By contrast, after connections have been developed to perform the task directly, there is no correspondingly stable pattern formed during the performance of the task. Thus the loss of conscious phenomenology with expertise can be understood naturally.

On this account, the sequentiality of the rule interpretation process is not built into the architecture; rather it is a consequence of the fact that we can follow only one instruction at a time. Connectionist memories have the capability to retrieve a single stored item, and here this is necessary to avoid asking the linguistic interpreter to simultaneously interpret more than one instruction.

It is interesting to note that the preceding analysis does not require that the "rules" be linguistic; any notational system that can be appropriately interpreted would do. Another type of "rule" is a series of musical pitches; a memorized collection of such rules allows a musician to play a tune by "conscious rule interpretation." With practice the need for conscious control goes away. Since pianists learn to interpret several notes simultaneously, the present account suggests that pianists might be able to apply more than one musical rule at a time (provided their memory for the rules can simultaneously recall more than one rule). A symbolic account of such conscious rule interpretation would involve something like a production system capable of firing multiple productions simultaneously.

Finally it should be noted that even if the memorized rules are assumed to be linguistically coded, the preceding analysis is uncommitted about the form the rules take in memory: phonological, orthographic, semantic, or whatever.

4.2. Symbolic *vs.* subsymbolic implementation of rule interpretation

The (approximate) implementation of the conscious rule interpreter in a subsymbolic system has both advantages and disadvantages relative to an (exact) implementation in a von Neumann machine.

The main disadvantage is that subconceptual representation and interpretation of linguistic instructions is very difficult and we can't now actually do it. Most existing subsymbolic systems simply don't use rule interpretation.[7] Thus they miss out on all the advantages of listed in (1). They can't take advantage of rules to check the results produced by the intuitive processor. They can't bootstrap their way into a new domain using rules to generate their own experience: they must have a teacher generate it for them.[8]

There are several advantages of a subconceptually implemented rule interpreter. The intuitive processor and rule interpreter are highly integrated, with broad-band communication between them. Understanding how this communication works should allow design of efficient hybrid symbolic/subsymbolic systems with effective communication between the processors. A principled basis is provided for studying how rule-based knowledge leads to intuitive knowledge. Perhaps most interesting, in a subsymbolic rule interpreter, the process of rule selection is intuitive! Which rule is reinstantiated in memory at a given time is the result of the associative retrieval process, which has many nice properties. The "best match" to the productions' conditions is quickly computed, and even if no match is very good, a rule can be retrieved. The selection process can be quite context-sensitive.

An integrated subsymbolic rule interpreter/intuitive processor in principle offers the advantages of both kinds of processing. Imagine such a system creating a mathematical proof. The intuitive processor would suggest goals and steps, and the rule interpreter would verify the validity of proposals. The serial search through the space of possible steps that is necessary in a purely symbolic approach is replaced by intuitive generation of possibilities. Yet the precise adherence to strict inference rules that is demanded by the task can be enforced by the rule interpreter; the creativity of intuition can be exploited while its unreliability can be controlled.

4.3. Two kinds of knowledge; one medium

Most existing subsymbolic systems perform tasks without serial rule interpretation: patterns of activity representing inputs are directly transformed (possibly through multiple layers of units) to patterns of activity representing outputs. The connections that mediate this transformation represent a form of task knowledge that can be applied with massive parallelism: I will call it *P-knowledge*. For example, the P-knowledge in a native speaker encodes lexical, morphological, syntactic, semantic, and pragmatic constraints in such form that all these constraints can be satisfied in parallel during comprehension and generation.

The connectionist implementation of sequential rule interpretation described above displays a second form that knowledge can take in a subsymbolic system. The stored activity patterns that represent rules also constitute task knowledge: call it *S-knowledge*. Like P-knowledge, S-knowledge is imbedded in connections: the connections that enable part of a rule to reinstantiate the entire rule. Unlike P-knowledge, S-knowledge cannot be used massively in parallel. For example, a novice speaker of some language cannot satisfy the constraints contained in two memorized rules simultaneously; they must be serially reinstantiated as patterns of activity and separately

interpreted. Of course the connections responsible for reinstantiating these memories operate in parallel, and indeed these connections contain within them the potential to reinstantiate either of the two memorized rules. But these connections are so arranged that *only one rule at a time* can be reinstantiated. The retrieval of a single rule is a parallel process, but the satisfaction of the constraints contained in the two rules is a serial process. After considerable experience, P-knowledge is created: connections that can *simultaneously satisfy* the constraints represented by the two rules.

P-knowledge is considerably more difficult to create than S-knowledge. To encode a constraint in connections so that it can be satisfied in parallel with thousands of others is no easy task. Such an encoding can only be learned through considerable experience in which that constraint has appeared in many different contexts, so that the connections enforcing the constraint can be tuned to operate in parallel with those enforcing a wide variety of other constraints. S-knowledge can be much more rapidly acquired (once the linguistic skills on which it depends have been encoded into P-knowledge, of course). Simply reciting a verbal rule over and over will usually suffice to store it in memory, (at least for a few days).

That P-knowledge is so highly context-dependent while the rules of S-knowledge are essentially context-free is an important computational fact underlying many of the psychological explanations offered by subsymbolic models. Consider, for example, Rumelhart and McClelland's (1986) model of the U-shaped curve for past-tense production in children. The phenomenon is striking: a child is observed using *goed* and *wented* when at a much younger age *went* was reliably used. This is surprising because we are prone to think that such linguistic abilities rest on knowledge that is encoded in some context-free form such as "the past tense of *go* is *went*." Why should a child *lose* such a rule once acquired? A traditional answer invokes the acquisition of a different context-free rule, like "the past tense of *x* is *x+ed*" which for some reason takes precedence. The point here, however, is that *there is nothing at all surprising about the phenomenon when the underlying knowledge is assumed to be context-dependent and not context-free.* The young child has a small vocabulary of largely irregular verbs. The connections that implement this P-knowledge are capable of reliably producing the large pattern of activity representing *went*, as well as those representing a small number of other past-tense forms. Informally we can say that the connections producing *went* do so *in the context of the other vocabulary items* that are also stored in the same connections. There is no guarantee that these connections will produce *went* in the context of a different vocabulary. As the child acquires additional vocabulary items, most of which are regular, the context radically changes. Connections that were, so to speak, perfectly adequate for creating *went* in the old context now have to work in a context where very strong connections are trying to create forms ending in *–ed*; these "old connections" are not up to the new task. Only through extensive experience trying to produce *went* in the new context of many regular verbs can the "old" connections be modified to work in the new context. (In particular, strong new connections must be added that, when the input pattern is that for *go*, cancel the *–ed*; these were not needed before.)

These observations about context-dependence can also be framed in terms of inference. If we choose to regard the child as using knowledge to in some sense "infer" the correct answer *went*, then we can say that after the child has added more knowledge (about new verbs), the ability to make the (correct) inference is lost. In this sense the child's inference process is *non-monotonic*—perhaps this is why we find the phenomenon surprising. As will be discussed in Section 6, non-monotonicity is a fundamental property of subsymbolic inference.

To summarize:

(17) a. Knowledge in subsymbolic systems can take two forms, both resident in the connections.
b. The knowledge used by the conscious rule interpreter lies in connections that reinstantiate patterns encoding rules; task constraints are coded in context-free rules and satisfied serially.
c. The knowledge used in intuitive processing lies in connections that constitute highly context-sensitive encodings of task constraints that can be satisfied with massive parallelism.
d. Learning such encodings requires much experience.

5. Subsymbolic definition of cognitive systems and some foundational issues

In order for the subconceptual level to rightly be viewed as a level for practicing cognitive science, it is necessary that the principles formulated at this level truly be principles of cognition. Since subsymbolic principles are neither conceptual-level nor neural-level principles, it is not immediately apparent what kind of cognitive principles they might be. The structure of subsymbolic models is that of a dynamical system; in what sense do these models embody principles of cognition rather than principles of physics?

What distinguishes those dynamical systems that are cognitive from those that are not? A crucial property of cognitive systems is that over a wide variety of environments they can maintain, at an adequately constant level, the degree to which a significant number of conditions are met: their "survival needs," "drives," "desires," or whatever you may prefer to call them—I'll use the term "goal conditions." A river, for example, is a complex dynamical system that responds sensitively to its environment, but about the only "goal" that it can be regarded as satisfying over a large range of environments is "going down hill." A cockroach manages, for an annoyingly amazing range of environments, to keep its body temperature, its nutritive intake, its reproductive demands, its oxygen intake, even its probability of getting smashed, all within a relatively narrow band. The repertoire of conditions that *people* can keep satisfied, and the range of environments under which this relative constancy can be maintained, provides a measure worthy of the human cognitive capacity.

(18) **Cognitive system:**
A cognitive system can, under a wide variety of environmental conditions, maintain a large number of goal conditions. The greater the repertoire of goals and variety of tolerable conditions, the greater the cognitive capacity of the system.

The issue of complexity is crucial here. A river (or a thermostat) fails to be a cognitive dynamical system only because it cannot satisfy a *large* range of goals under a *large* range of conditions.[9] Complexity is largely what distinguishes the dynamical systems studied in the subsymbolic paradigm from those traditionally studied in physics. Connectionist dynamical systems have great complexity: the information content in their weights is very high. Studying the extent to which a connectionist dynamical system can achieve complex goals in complex environments requires grappling with complexity in dynamical systems in a way that is traditionally avoided in physics. In cognitive modeling, many of the essential questions concern the detailed dynamics of a particular pattern of activation in a system with a particular initial state and a particular set of interaction strengths that are highly nonhomogeneous. This is like asking a physicist: "Suppose we have a gas with 10,000 particles with the following 10,000 different masses and the following 500,000 different forces between them. Suppose we start them at rest in the following 10,000 positions. What are the trajectories of the following 20 particles?" This is indeed a question about a dynamical system, and is, in a sense, a question of physics. But it is just the kind of question physicists tend to avoid at all costs. Our physicist is likely to compute the mean collision times for the particles assuming equal masses, random starting positions, and uniformly random interactions, and say "if that isn't good enough, then take your question to a computer."

Nonetheless, physics has valuable concepts and techniques to contribute to the study of connectionist dynamical systems. Insights from physics have already proved important in various ways in the subsymbolic paradigm (eg. Hinton & Sejnowski 1983a; Sejnowski 1976; Smolensky 1983).

Various subsymbolic models have addressed various goals and environments. A very general goal that is of particular importance is:

(19) *The prediction goal:* Given some partial information about the environmental state, correctly infer missing information.

What is maintained here is the degree of match between predicted values and the actual values for the unknowns. Maintenance of this match over the wide range of conditions found in a complex task environment is a difficult task. Special cases of this task include predicting the depth of an object from retinal images, or the future location of a moving object, or the change in some aspects of an electric circuit given the changes in other aspects, or the unstated propositions in a text. The prediction goal is obviously an important one, all the more because it can serve

so many other goals: accurate prediction of the effects of actions enables selection of those leading to desired effects.

A closely related goal is

(20) *The prediction-from-examples goal*: Given more and more examples of states from an environment, achieve the prediction goal in that environment with increasing accuracy.

For the prediction goal we ask: What *inference procedures* and *knowledge* about an environment must a dynamical system possess to be able to predict that environment? For the prediction-from-examples goal we go further and ask: What *learning procedures* must a dynamical system possess to be able to acquire the necessary knowledge about an environment from examples?

The goals of prediction and prediction-from-examples are the subject of many of the principles of the subsymbolic paradigm. These are indeed cognitive principles. These principles will be taken up in the next section; first, however, I would like to consider some implications of this characterization of a cognitive system for certain foundational issues: semantics, rationality, the constituent structure of mental states, and propositional attitudes. The hypotheses formulated in the remainder of this section are quite preliminary.

5.1. Semantics and rationality in the subsymbolic paradigm

The subsymbolic definition of a cognitive system (18) intrinsically binds cognitive systems both to states of the environment and to goal conditions. It therefore takes a position on the question: How do states of a subsymbolic system get their meanings and truth conditions? A preliminary answer is the following.

(21) **Subsymbolic semantics:**
A cognitive system adopts various internal states in various environmental conditions. To the extent that the cognitive system meets its goal conditions in various environmental conditions, its internal states are *veridical representations* of the corresponding environmental states, with respect to the given goal conditions.

For the prediction goal, for example, a state of the subsymbolic system is a veridical representation of the current environmental state to the extent that it leads to correct predictions.

According to hypothesis (21), it is not necessarily possible to localize a failure of veridical representation. Any particular state is part of a large causal system of states, and failures of the system to meet goal conditions cannot in general be localized to any particular state or state component.[10] In subsymbolic systems, this *assignment of blame* problem is a difficult one, and it makes programming subsymbolic models by hand very tricky. Solving the assignment of blame problem is one of the central accomplishments of the automatic network programming procedures: the learning procedures of the subsymbolic paradigm.

The definition (18) of cognitive systems clearly relates to rationality as well. How can one build a rational machine? How can internal processes (eg. inference) be guaranteed to preserve veridical semantic relationships (eg. be truth preserving)? These questions now become: How can the connection strengths be set so that the subsymbolic system will meet its goal conditions? Again, this is a question answered by the scientific discoveries of the subsymbolic paradigm: particular procedures for programming machines to meet certain goals—especially learning procedures to meet adaptation goals like prediction-from-examples.

Let me contrast this subsymbolic approach to veridicality with a symbolic approach to truth preservation offered by Fodor (19xx). In the context of model-theoretic semantics for a set of symbolic formulae, proof theory provides a set of symbol manipulations (rules of inference) guaranteed to preserve truth conditions. Thus if an agent possesses knowledge in the symbolic form $p \rightarrow q$ and additional knowledge p, then by syntactic operations the agent can produce q; proof theory guarantees that the truth conditions of the agent's knowledge (or, if you prefer, beliefs) has not changed.

While this account may explain the tautological inference of q, it does not explain what is most interesting from a cognitive point of view: *How did the agent know that $p \rightarrow q$ in the first place*? The ability of the agent to derive q depends entirely on the implication having somehow already been given to the agent. This account trivializes the veridicality problem by treating all inference as tautological and ignoring crucial questions such as: How can a cognitive system be put in a novel environment and learn to create veridical internal representations that allow valid inferences about that environment so that goal conditions can be satisfied? How can it pick up information from its environment? These questions are explicitly addressed in the subsymbolic paradigm, and are answered by the learning procedures.

Note that in the subsymbolic case, the internal processing mechanisms (which we can call "inference" if we like) do not, of course, directly causally depend on the environmental state that may be internally represented or on the veridicality of that representation. They are just as "formal" in that sense as syntactic symbol manipulations. The fact that a subsymbolic system can generate veridical representations of the environment (eg., make valid predictions) is a result of extracting information from the environment and internally coding it in its weights through a learning procedure.

While learning procedures may play the most interesting role in the theory of subsymbolic semantics, for certain subsymbolic systems there is also a direct analog of the proof theory account. The role of logical inference is played by *statistical inference*. By explicitly formalizing tasks like prediction as statistical inference tasks, it is possible to prove for appropriate systems that subsymbolic computation is valid in a sense directly comparable to symbolic proof. Further discussion of this point, which will appear in section 7.1, must await further examination of the computational framework of the subsymbolic paradigm, which is the subject of section 6.

5.2. Constituent structure of mental states

It has been argued (eg. Fodor 1975; Pylyshyn 1984) that mental states must have constituent structure. An internal representation in a subsymbolic system is a pattern of activity which has constituent structure that can be analyzed at both the conceptual and subconceptual levels. At the conceptual level, a state contains many constituent subpatterns which have conceptual interpretation. A state resulting from the processing of the phrase *cup with coffee* will presumably contain constituent subpatterns that can be analyzed as separately representing *cup* and *coffee*. But such a decomposition differs in two important respects from the decomposition of the linguistic structure *cup with coffee* itself into its three constituents *cup*, *coffee*, and *with*. First, the constituents are activity vectors that are *superimposed* on each other, not symbols that are concatenated. This has great technical significance, and is a manifestation of profound differences in the symbolic and subsymbolic computational frameworks, as will be discussed in the next section. The second point, more significant from the philosophical point of view, is that *the decomposition into constituents is highly approximate*. There is no precise way of specifying exactly the constituent representing *coffee* in the interpretation of *cup with coffee*. In the interpretation of the phrases *cup with coffee*, *can with coffee*, *tree with coffee*, and *man with coffee*, the constituents representing *coffee* (to the approximate extent to which they can be defined) are all significantly different. These constituents represent *coffee* in varying contexts; these vectors are not identical but possess a rich structure of commonalities and differences (a "family resemblance," so to speak). The commonalities are directly responsible for the common processing implications of the interpretations of these various phrases, so the *approximate* equivalence of the "*coffee* vectors" across contexts plays a functional role in processing that is very close to that played by the *exact* equivalence of the *coffee* tokens across different contexts in a symbolic processing system.

That constituents analyzed at the subconceptual level are important to mental descriptions in the subsymbolic paradigm can be described in another way. For this purpose let's call each "*coffee* vector" the *symbol* for *coffee* in the given context. Then we can say that the context alters the *internal structure* of the symbol; the activities of the subconceptual units that comprise the symbol—its *subsymbols*—change across contexts. In the symbolic paradigm, a symbol is effectively contextualized by *surrounding it* with other symbols in some larger structure. In other words:

(22) **Symbols and context dependence:**
In the symbolic paradigm, the context of a symbol is manifest *around* it, and consists of *other symbols*; in the subsymbolic paradigm, the context of a symbol is manifest *inside* it, and consists of *subsymbols*.

(See Hofstadter 1979, 1985.)

5.3. Propositional Attitudes

Several of the points raised so far can be assembled into a preliminary, partial account of propositional attitudes within the subsymbolic paradigm.

Among the most important aspects of the environment that we must as cognitive systems predict is the behavior of other cognitive systems. For people (and animals) with which we have a great deal of experience (eg. ourselves), our intuitive processors are often capable of making quite reliable predictions. We often cannot provide good verbal accounts for such predictions. To predict the behavior of less well-known agents, we can use our intuitive knowledge of animals or people of various sorts, but for more than very general predictions we need information that further specifies them as cognitive systems: their particular goals and their relations to the environment. This information usually comes in verbal form (although we certainly make ample use of appearance, manner, and other nonverbal sources of information when it is available). The verbal expression of an agent's goals is relatively straightforward: "Grandma Jerry likes good opera." The relation of the agent to the environment is a bit more complex. An important discovery of childhood is that to predict an agent's behavior in a given environment, it is often necessary to transform that environment to another one that reflects the agent's different perspective. These perspectives come in straightforward forms such as visual perspectives and more subtle forms such as political perspectives. Verbal information that enables more abstract transformation of perspectives often takes the form of *belief ascriptions*: "Grandma Jerry believes there is no good opera in the West." Now I can use my intuition about cognitive systems to predict Grandma Jerry's behavior, making use of his alleged goals after I have transformed my representation of the Sante Fe opera into one that is allegedly appropriate for Grandma Jerry. The development of this intuitive skill rests on the daily experience we each get with it, including the relating of our own behavior to our own verbal descriptions of our goals and beliefs.

An ascription of belief is a verbal expression that allows us to transform our representation of the environment into one appropriate for prediction of another agent's behavior; it relates the agent to some verbally expressible aspect of an environment. The form of this expression characteristic of beliefs is a statement about the environment that is true in the agent's representation: *A believes that P holds in the environment.* Thus to ascribe a belief is to assert a relation between an agent *A* and a verbal proposition *P*. But on this subsymbolic account, since typically *P* relates information with which we have much experience, the processing of this verbal information is typically by the intuitive processor, and does not involve symbolic manipulation. (Of course, if I say "A believes that zorch is grue, and A believes all non-bleen things are non-grue, and A desires bleen things; what do you think A will do if I offer her some zorch?", then the conscious rule interpreter—and possibly even pencil and paper!—will have an atypically large role to play.)

6. Computation at the subconceptual level

Hypothesis (7a) offers a brief characterization of the connectionist architecture assumed at the subconceptual level by the subsymbolic paradigm. It is time to bring out the computational principles implicit in that hypothesis.

6.1. Continuity

According to (7a), a connectionist dynamical system has a continuous space of states and changes state continuously in time. The first order of business is to motivate this continuity condition, reconcile it with some apparent counterexamples, and point out some of its implications.

Within the symbolic paradigm it is natural to formalize a number of cognitive processes in quite discrete terms:

(23) a. Discrete memory locations, in which items are stored without mutual interaction.
b. Discrete memory storage and retrieval operations, in which an entire item is stored or retrieved in a single, atomic (primitive) operation.
c. Discrete learning operations, in which new rules suddenly come into being whole.
d. Discrete inference operations, in which conclusions suddenly come into being whole.
e. Discrete categories, to which items either belong or do not.
f. Discrete production rules, with conditions that are either satisfied or not, and actions that either execute or do not execute.

Cognitive behavior often shows much less discreteness than this. Indeed, cognition seems to be a richly interwoven fabric of continuous and discrete processes. One way to model this interplay is to posit separate discrete and continuous processors in interaction. Some theoretical problems with this move were mentioned in Section 4, where a unified formalism was advocated. It is difficult to introduce a hard separation between the soft and the hard components of processing. An alternative is to adopt a fundamentally symbolic approach, but to soften various forms of discreteness by hand. For example, the degree of match to conditions of production rules can be given numerical values, productions can be given strengths, interactions between separately stored memory items can be put in by hand, and so on (eg., see Anderson, 1983).

The subsymbolic paradigm offers another alternative. All the discrete features of (23) are neatly swept aside in one stroke by adopting a fundamentally continuous framework. Then, when the continuous system is analyzed at a higher level, various aspects of discreteness *emerge naturally*. These aspects of hardness are intrinsically imbedded in a fundamentally soft system.

It may appear that the continuous nature of subsymbolic systems is contradicted by the fact that it is easy to find in the literature models that are quite within the spirit of the subsymbolic paradigm but which have neither continuous state spaces nor continuous dynamics: for example, models having units with binary values that jump discretely on the ticks of a discrete clock (eg., the Boltzmann machine, Hinton & Sejnowski, 1983a, Ackley, Hinton & Sejnowski, 1985; harmony theory, Smolensky 1983, 1986a). I will now argue that these models should be viewed as discrete simulations of an underlying continuous model, considering first discretization of time and then discretization of the units' values.

Dynamical systems evolving in continuous time are nearly always simulated on digital computers by discretizing time. Since subsymbolic models have almost always been simulated on digital computers, it is no surprise that they too have been so simulated. The equations defining the dynamics of the models can be more easily understood by most cognitive scientists if the differential equations of the underlying continuous dynamical system are avoided in favor of the discrete-time approximations that actually get simulated. But these discrete-time equations are widely recognized as approximations for the underlying differential equations.

When subsymbolic models employ binary-valued units, these values are best viewed not as symbols like *T* and *NIL* that are used for conditional branching tests, but as numbers (not numerals!) like 1 and 0 that are used for numerical operations (eg., multiplication by weights, summation, exponentiation). These models are formulated in such a way that they are perfectly well-defined for continuous values of the units. Discrete numerical unit values is no more than a simplification that is sometimes convenient. For example, in both harmony theory and the Boltzmann machine, discrete units have typically been used because: (1) discrete units simplify both analysis and simulation; (2) for the quadratic harmony or energy functions that are being optimized, it can be proved that no optima are lost by simplifying to binary values; (3) these models' stochastic search has a "jumpy" quality to it anyway.

As historical evidence that underlying subsymbolic models are continuous systems, it is interesting to note several occasions where the theoretical conditions that license the discrete approximation have changed and the models have reverted to continuous values. In the harmony/energy optima model, when the jumpy stochastic search was replaced by a smooth deterministic one (Rumelhart, Smolensky, McClelland & Hinton 1986), the units were changed to continuous ones. Alternatively, if the original harmony/Boltzmann approach is extended to include

non-quadratic harmony/energy functions, non-binary optima appear, so again one switches to continuous units (Smolensky, in progress). The most dramatic evidence is a case where switching from discrete to continuous units enabled a revolution in subsymbolic learning theory. In their classic, *Perceptrons*, Minsky and Papert (1969) exploited more or less discrete mathematical methods that were compatible with the choice of binary units. They were incapable of analyzing any but the simplest learning networks. By changing the discrete threshold function of perceptrons to a smooth, differentiable curve, and thereby defining continuously-valued units, Rumelhart, Hinton and Williams (1986) were able to apply continuous analytic methods to more complex learning networks. The result has been a quantum leap in the power of subsymbolic learning.

The final point is a foundational one. The theory of discrete computation is quite well understood. If there is any new theory of computation implicit in the subsymbolic approach, it is likely to be a result of a fundamentally different, continuous formulation of computation.

It must be emphasized that the discrete/continuous distinction is not to be clearly understood by looking at simulations. Discrete and continuous machines can of course simulate each other. The claim here is that the most analytically powerful descriptions of subsymbolic models are continuous ones while those of symbolic models are not.

This has profound significance because it means that many of the concepts used to understand cognition in the subsymbolic paradigm come from the category of continuous mathematics, while those used in the symbolic paradigm come nearly exclusively from discrete mathematics. Concepts from physics, from the theory of dynamical systems, are at least as likely to be important as concepts from the theory of digital computation. And analog computers, both electronic and optical, provide natural implementation media for subsymbolic systems (eg. Anderson 1986; Cohen 1986).

6.2. Subsymbolic computation

An important instance of the continuous/discrete mathematics contrast that distinguishes subsymbolic and symbolic computation is found in inference. A natural way to look at the knowledge stored in connections is to view each connection as a *soft constraint*. A positive ("excitatory") connection from unit *a* to unit *b* represents a soft constraint to the effect that if *a* is active, then *b* should be too. A negative ("inhibitory") connection represents the opposite constraint. The numerical magnitude of a connection represents the strength of the constraint.

Formalizing knowledge in soft constraints rather than hard rules has important consequences. Hard constraints have consequences singly; they are context-free rules that can be applied separately, sequentially. But *soft constraints have no implications singly;* any one can be overridden by the others. It is only the *entire set* of soft constraints that has any implications. Inference must be a cooperative process, like the parallel relaxation processes typically found in subsymbolic systems. Furthermore, adding additional soft constraints can repeal conclusions that were formerly valid. Subsymbolic inference is fundamentally non-monotonic.

One way of formalizing soft constraint satisfaction is in terms of statistical inference. In certain subsymbolic systems, the soft constraints can be identified as statistical parameters and the activation passing procedures can be identified as statistical inference procedures (Hinton & Sejnowski 1983b; Geman & Geman 1984; Smolensky 1986a; see also Shastri 1985; Pearl 1985). This identification is usually rather complex and subtle, and is usually not simply a matter of identifying the strength of the connection between two units with the correlation between their activity. An important goal is to determine how statistical and other formal theories of continuous (as opposed to logical) inference can be employed to mathematically elucidate the inference found in other subsymbolic systems.

To sum up:

(24) a. Knowledge in subsymbolic computation is formalized as a large set of soft constraints.

b. Inference with soft constraints is fundamentally a parallel process.
c. Inference with soft constraints is fundamentally non-monotonic.
d. Certain subsymbolic systems can be identified as employing statistical inference.

7. Conceptual-level descriptions of intuition

The previous section concerned computation in subsymbolic systems analyzed at the subconceptual level, the level of units and connections. In this final section I consider analyses of subsymbolic computation at the higher, conceptual level. Section 4 discussed subsymbolic modeling of conscious rule interpretation; here, I consider subsymbolic models of intuitive processes. I will elaborate the point foreshadowed in Section 3: conceptual-level descriptions of aspects of subsymbolic models of intuitive processing roughly approximate symbolic accounts. The picture that emerges is of a symbiosis between the symbolic and subsymbolic paradigms: the symbolic paradigm offers concepts for better understanding subsymbolic models, and those concepts are in turn illuminated with a fresh light by the subsymbolic paradigm.

7.1. The Best Fit Principle

The notion that each connection represent a soft constraint can be formulated at a higher level:

(25) **The Best Fit Principle:**
Given an input, a subsymbolic system outputs a set of inferences that, as a whole, give a best fit to the input, in a statistical sense defined by the statistical knowledge stored in the system's connections.

In this vague form, this principle can be regarded as a desideratum of subsymbolic systems. But it is exactly true in a precise sense, at least in an idealized limit, for the class of connectionist dynamical systems that have been studied in harmony theory (Riley & Smolensky 1984; Smolensky 1983, 1984a, 1984b, 1986a, 1986b, 1986c).

To render the Best Fit Principle precise, it is necessary to provide precise definitions of "inferences," "best fit" and "statistical knowledge stored in the system's connections." This is done in harmony theory, where the central object is the harmony function H which measures, for any possible set of inferences, the goodness of fit to the input with respect to the soft constraints stored in the connection strengths. The set of inferences with the largest value of H, i.e. highest harmony, is the best set of inferences, with respect to a well-defined statistical problem.

Harmony theory basically offers three things. It gives a mathematically precise characterization of the prediction-from-examples goal as a statistical inference problem. It tells how the prediction goal can be achieved using a connectionist network with a certain set of connections. And it gives a procedure by which the network can learn the correct connections with experience, thereby satisfying the prediction-from-examples goal.

The units in harmony networks are stochastic units: the differential equations defining the system are stochastic. There is a system parameter called the *computational temperature* that governs the degree of randomness in the units' behavior: it goes to zero as the computation proceeds. (The process is *simulated annealing*, as in the Boltzmann machine: Ackley, Hinton & Sejnowski 1985; Hinton & Sejnowski, 1983a, 1983b, 1986. See Rumelhart, McClelland, & the PDP Research Group, 1986, p. 148, and Smolensky, 1986a, for the relations between harmony theory and the Boltzmann machine.)

7.2. Productions, sequential processing, and logical inference

A simple harmony model of expert intuition in qualitative physics was described in Riley and Smolensky (1984) and Smolensky (1986a, 1986c). The model answers questions like "what happens to the voltages in this circuit if I increase this resistor?" Higher level descriptions of this subsymbolic problem-solving system illustrate several interesting points.

It is possible to identify *macro-decisions* during the system's solution of a problem; these are each the result of many individual micro-decisions by the units of the system, and each amounts to a large-scale commitment to a portion of the solution. These macro-decisions are approximately like the firing of production rules. In fact, these "productions" "fire" in essentially the same order as in a symbolic forward-chaining inference system. One can measure the total amount of order in the system, and see that there is a qualitative change in the system when the first micro-decisions are made: the system changes from a disordered phase to an ordered one.

It's a corollary of the way this network embodies the problem domain constraints, and the general theorems of harmony theory, that the system, when given a well-posed problem, and infinite relaxation time, will always give the correct answer. So under that idealization, the *competence* of the system is described by *hard* constraints: Ohm's Law, Kirchoff's Law. It's as though it had those laws written down inside it. However, as in all subsymbolic systems, the *performance* of the system is achieved by satisfying a large set of *soft* constraints. What this means is that if we go outside of the ideal conditions under which hard constraints seem to be obeyed, the illusion that the system has hard constraints inside is quickly dispelled. The system can violate Ohm's Law if it has to, but if it doesn't have to violate the law, it won't. Thus, *outside the idealized domain of well-posed problems and infinite processing time, the system gives sensible performance.* It isn't brittle the way that symbolic inference systems are. If the system is given an ill-posed problem, it satisfies as many constraints as possible. If it is given inconsistent information, it doesn't fall flat, and deduce anything. If it is given insufficient information, it doesn't just sit there and deduce nothing. Given finite processing time, the performance degrades gracefully as well. So the competence/performance distinction can be addressed in a sensible way.

Returning to a physics level analogy introduced in Section 3, this "quantum" system appears to be "Newtonian" under the proper conditions. A system that has, at the micro-level, soft constraints, satisfied in parallel, *appears* at the macro-level, under the right circumstances, to have hard constraints, satisfied serially. But it doesn't *really*, and if you go outside the "Newtonian" domain, you see that it's really been a "quantum" system all along.

7.3. Conceptual-level spreading activation

In Section 5.2 it was pointed out that states of a subsymbolic model can be approximately analyzed as superpositions of vectors with individual conceptual-level semantics. It is possible to approximately analyze connectionist dynamical systems at the conceptual level, using the mathematics of the superposition operation. If the connectionist system is purely linear (so that the activity of each unit is precisely a weighted sum of the activities of the units giving it input), it can easily be proved that the higher level description obeys formal laws of just the same sort as the lower level: the subconceptual and conceptual levels are *isomorphic*. Linear connectionist systems are however of limited computational power, and most interesting connectionist systems are nonlinear. However, most of these are in fact *quasi*-linear: each unit *combines* its inputs linearly even though the effects of this combination on the unit's activity is nonlinear. Further, the problem-specific *knowledge* in such systems is in the combination weights, i.e. the *linear part* of the dynamical equations; and in learning systems it is generally only these linear weights that adapt. For these reasons, even though the higher level is not isomorphic to the lower level in nonlinear systems, there are senses in which the higher level *approximately* obeys formal laws similar to the lower level. (For details, see Smolensky 1986b.)

The conclusion here is a rather different one from the preceding section, where we saw how there are senses in which higher level characterizations of certain subsymbolic systems approximate productions, serial processing, and logical inference. What we see now is that there are also senses in which the laws approximately describing cognition at the conceptual level are *activation-passing laws* like those at the subconceptual level, but operating between units with individual conceptual semantics. Such semantic level descriptions of mental processing (which include *local* connectionist models; see note 4) have been of considerable value in cognitive science. We can now see how these "spreading activation" accounts of mental processing fit into the subsymbolic paradigm.

7.4. Schemata

The final conceptual-level notion I will consider is that of the *schema* (eg., Rumelhart, 1980). This concept goes back at least to Kant (1787/1963) as a description of mental concepts and mental categories. Schemata appear in many AI systems in the forms of frames, scripts, or similar structures: they are prepackaged bundles of information that support inference in stereotyped situations.

I will very briefly summarize work on schemata in connectionist systems reported in Rumelhart, Smolensky, McClelland & Hinton (1986; see also Feldman, 1981, and Smolensky, 1986a, 1986c). This work addressed the case of schemata for rooms. Subjects were asked to describe some imagined rooms using a set of 40 features like has-ceiling, has-window, contains-toilet, and so on. Statistics were computed on this data and these were used to construct a network containing one node for each feature, and containing connections computed from the statistical data.

This resulting network can perform inference of the same general kind as that carried out by symbolic systems with schemata for various types of rooms. The network is told that some room contains a ceiling and an oven; the question is, what else is likely to be in the room? The system settles down into a final state, and the inferences contained in that final state are that the room contains a coffee cup but no fireplace, a coffee pot but no computer.

The inference process in this system is simply one of greedily maximizing harmony. To describe the inference of this system on a higher level, we can examine the global states of the system in terms of their harmony values. How internally consistent are the various states in the space? It's a 40-dimensional state space, but various 2-dimensional subspaces can be selected and the harmony values there can be graphically displayed. The harmony landscape has various peaks; looking at the features of the state corresponding to one of the peaks, we find that it corresponds to a prototypical bathroom; others correspond to a prototypical office, and so on for all the kinds of rooms subjects were asked to describe. There are no *units* in this system for bathrooms or offices: there are just lower-level descriptors. The prototypical bathroom is a pattern of activation, and the system's recognition of its prototypicality is reflected in the harmony peak for that pattern. It is a consistent, "harmonious" combination of features: better than neighboring points like one representing a bathroom without a bathtub, which has distinctly lower harmony.

During inference, this system climbs directly uphill on the harmony landscape. When the system state is in the vicinity of the harmony peak representing the prototypical bathroom, the inferences it makes are governed by the shape of the harmony landscape there. This shape is like a "schema" that governs inferences about bathrooms. (In fact, harmony theory was created to give a connectionist formalization of the notion of schema; see Smolensky, 1984b, 1986a, 1986c.) Looking closely at the harmony landscape we can see that the terrain around the "bathroom" peak has many of the properties of a bathroom schema: variables and constants, default values, schemata imbedded inside of schemata, and even cross-variable dependencies, which are rather difficult to incorporate into symbolic formalizations of schemata. The system behaves as though it had schemata for bathrooms, offices, etc., even though they are not "really there" at the fundamental level: these schemata are strictly properties of a higher-level description. They are informal, approximate descriptions—one might even say they are merely metaphorical descriptions—of an inference process too subtle to admit such high-level descriptions with great precision. Even though these schemata may not be the sort of object on which to base a formal model, nonetheless they *are* useful descriptions that help us understand a very complex inference system.

7.5. Conclusion

The view of symbolic structures that emerges from viewing them as entities of high-level descriptions of dynamical systems is quite different from the view provided by the symbolic paradigm.

(26) a. Macro-inference is not a process of firing a symbolic production but rather of qualitative state change in a dynamical system, such as a phase transition.

b. Schemata are not large symbolic data structures but rather the potentially quite intricate shapes of harmony maxima.
c. Categories (it turns out) are attractors in connectionist dynamical systems: states that "suck in" to a common place many nearby states, like peaks of harmony functions.
d. Categorization is not the execution of a symbolic algorithm but the continuous evolution of the dynamical system, the evolution that drives states into the attractors, to maximal harmony.
e. Learning is not the construction and editing of formulae, but the gradual adjustment of connection strengths with experience, with the effect of slowly shifting harmony landscapes, adapting old and creating new concepts, categories, schemata.

The heterogeneous assortment of high-level mental structures that have been embraced in this section suggests that the conceptual level lacks formal unity. This is just what one expects of approximate higher-level descriptions, which, capturing different aspects of global properties, can have quite different characters. According to the subsymbolic paradigm, the unity underlying cognition is to be found not at the conceptual level, but rather at the subconceptual level, where relatively few principles in a single formal framework lead to a rich variety of global behaviors.

Acknowledgements

I am indebted to Dave Rumelhart for several years of provocative conversations on many of these issues, and for a number of the ideas formulated here. Sincere thanks to Jerry Fodor and Zenon Pylyshyn for most instructive conversations. Detailed comments on an earlier draft from Mark Fanty and Dan Lloyd were very helpful, as were pointers from Kathleen Akins. I am especially grateful for the many insights that Rob Cummins and Denise Dellarosa have generously contributed to this paper.

This research has been supported by NSF grant IST-8609599 and by the Department of Computer Science and Institute of Cognitive Science at the University of Colorado at Boulder.

Footnotes

1. In elementary particle physics, *PTC* (or any permutation of the letters) refers to an operation in which left/right Parity, the direction of Time, and the sign of electric Charge are all simultaneously inverted (eg, Streater & Wightman 1964). There is an analogous reading of *PTC* as three simultaneous inversions that map the symbolic and subsymbolic approaches onto each other: the type of Psychological process most naturally modeled (rule application *vs.* intuitive processing), the semantic level at which the Task is modeled (conceptual *vs.* subconceptual), and the type of Computation employed in the model (discrete, symbolic *vs.* continuous, numerical). Since a cognitive model is just a computational system performing a task so as to model a psychological process, the two paradigms differ on all three of the dimensions intrinsic to the modeling enterprise.

2. In this paper, when *interpretation* is used to refer to a process, the sense intended is that of computer science: the process of taking a linguistic description of a procedure and executing that procedure.

3. Consider, for example, the connectionist symposium at the University of Geneva held Sept. 9, 1986. The advertised program featured Feldman, Minsky, Rumelhart, Sejnowski, and Waltz. Of these five researchers, three were major contributors to the symbolic paradigm for many years: consider Minsky 1975; Rumelhart 1975, 1980; and Waltz 1978, for example.

4. This is an issue that divides connectionist approaches. "Local connectionist models" (eg. Dell 1985; Feldman 1985; McClelland & Rumelhart 1981; Rumelhart & McClelland 1982; Waltz & Pollack 1985) accept (8), and often deviate significantly from (7a). This approach has been championed by the Rochester connectionists (see Feldman, Ballard, Brown & Dell 1985). Like the symbolic paradigm, this school favors simple semantics and more complex operations. The processors in their networks are usually more powerful than those allowed by (7); they are often rather like digital computers running a few lines of simple code. ("If there is a 1 on this input line then do X else do Y," where X and Y are quite different little procedures; eg., Shastri 1985.) This style of connectionism, quite different from the subsymbolic style, has much in common with the branch of traditional computer science that "parallelizes" serial algorithms by decomposing them into routines that can run in parallel, often with certain synchronization points built in. The grain size of the Rochester parallelism, while large compared to the subsymbolic paradigm, is small compared to standard parallel programming: the processors are allowed only a few internal states and allowed to transmit only a few different values (Feldman & Ballard, 1982).

5. As stated in the introduction, a large sample of research that by and large falls under the subsymbolic paradigm can be found in the books, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*: Rumelhart, McClelland & the PDP Research Group 1986; McClelland, Rumelhart & the PDP Research Group 1986. While this work has come to be labelled "connectionist," the term "PDP" was deliberately chosen to distinguish it from the localist approach which had previously adopted the name "connectionist" (Feldman and Ballard, 1982).

6. The phrase is Roger Schank's, in reference to "parallel processing." Whether he was referring to connectionist systems I do not know; in any event, I don't mean to imply that the grounds for his comment are addressed here.

7. A notable exception is Touretzky & Hinton 1985.

8. And when a network makes a mistake, it can be told the correct answer but it can't be told which rule it violated. Thus it must assign blame for its error in a very undirected way. It is quite plausible that the large amount of training currently required by subsymbolic systems could be significantly reduced if blame could be focussed by citing violated rules.

9. There is a trade-off between the number of goal conditions one chooses to attribute to a system and the corresponding range of tolerable conditions. Considering a large variety of environmental conditions for a river, there is only the "flow downhill" goal; by appropriately narrowing the class of conditions, one can increase the corresponding goal repertoire. A river can carry messages from A to B, if A and B are appropriately restricted. But a homing pigeon can meet this goal over a much greater variety of situations.

10. This problem is closely related to the localization of a failure of veridicality in a scientific theory. Pursuing the remarks of section 1.1, scientific theories can be viewed as cognitive systems, indeed ones having the prediction goal. Veridicality is a property of a scientific theory as a whole, gauged ultimately by the success or failure of the theory to meet the prediction goal. The veridicality of abstract representations in a theory derives solely from their causal role in the accurate predictions of observable representations.

References

Ackley, D.H., Hinton, G.E., & Sejnowski, T.J. (1985) A learning algorithm for Boltzmann machines. *Cognitive Science* 9: 147–169.

Anderson, D.Z. (1986) Coherent optical eigenstate memory. *Optics Letters* 11: 56–58.

Anderson, J.R. (1981) *Cognitive skills and their acquisition*. Erlbaum.

Anderson, J.R. (1983) *The architecture of cognition*. Harvard University Press.

Cohen, M.S. (1986) Design of a new medium for volume holographic information processing. *Applied Optics* 14: 2288–2294.

Crick, F. & Asanuma, C. (1986) Certain aspects of the anatomy and physiology of the cerebral cortex. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Dell, G.S. (1985) Positive feedback in hierarchical connectionist models: Applications to language production. *Cognitive Science* 9: 3-23.

Feldman, J.A. (1985) Four frames suffice: A provisional model of vision and space. *The Behavioral and Brain Sciences* 8: 265–289.

Feldman, J.A. (1981) A connectionist model of visual memory. In: *Parallel models of associative memory*, ed. G.E. Hinton & J.A. Anderson. Erlbaum.

Feldman, J.A. (1986) Neural representation of conceptual knowledge. Technical Report 189, Department of Computer Science, University of Rochester.

Feldman, J.A. & Ballard, D.H. (1982) Connectionist models and their properties. *Cognitive Science* 6: 205–254.

Feldman, J.A., Ballard, D.H., Brown, C.M., & Dell, G.S. (1985) Rochester connectionist papers: 1979–1985. Technical Report 172, Department of Computer Science, University of Rochester.

Fodor, J.A. (1975) *The language of thought*. Crowell.

Fodor, J.A. (19xx) Why there still needs to be a language of thought.

Geman, S., & Geman, D. (1984) Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6: 721–741.

Haugeland, J. (1978) The nature and plausibility of cognitivism. *Behavioral and Brain Sciences* 1: 215–226.

Hebb, D.O. (1949) *The organization of behavior*. Wiley.

Hinton, G.E. and Anderson, J.A. (Eds.) (1981) *Parallel models of associative memory*. Erlbaum.

Hinton, G.E., McClelland, J.L., & Rumelhart, D.E. (1986) Distributed representations. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Hinton, G.E. & Sejnowski, T.J. (1983a) Analyzing cooperative computation. *Proceedings of the Fifth Annual Conference of the Cognitive Science Society.*

Hinton, G.E. & Sejnowski, T.J. (1983b) Optimal perceptual inference. *Proceedings of the I.E.E.E. Conference on Computer Vision and Pattern Recognition.*

Hinton, G.E., & Sejnowski, T.J. (1986) Learning and relearning in Boltzmann machines. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Hofstadter, D.R. (1979) *Godel, Escher, Bach: An eternal golden braid.* Basic Books.

Hofstadter, D.R. (1985) Waking up from the Boolean dream, or, subcognition as computation. In: *Metamagical themas*, pp. 631–665. Basic Books.

Kant, E. (1787/1963) *Critique of pure reason.* N. Kemp Smith, trans.; 2nd ed. McMillan.

Larkin, J.H., McDermott, J., Simon, D.P., & Simon, H.A. (1980) Models of competence in solving physics problems. *Cognitive Science* 4: 317–345.

Lashley, K. (1950) In search of the engram. In: *Psychological Mechanisms in Animal Behavior*, Symposia of the Society for Experimental Biology, No. 4, pp. 454–483. Academic.

Lewis, C.H. (1978) *Production system models of practice effects.* Unpublished doctoral dissertation, University of Michigan.

McClelland, J.L. & Rumelhart, D.E. (1981) An interactive activation model of context effects in letter perception: Part 1. An account of the basic findings. *Psychological Review* 88: 375–407.

McClelland, J.L., Rumelhart, D.E., & the PDP Research Group. (1986) *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models.* MIT Press/Bradford Books.

Minsky, M. (1975) A framework for representing knowledge. In: *The psychology of computer vision*, ed. P.H. Winston, pp. 211-277. McGraw-Hill.

Minsky, M., & Papert, S. (1969) *Perceptrons.* MIT Press.

Pearl, J. (1985) Bayesian networks: A model of self-activated memory for evidential reasoning. *Proceedings of the 7th Conference of the Cognitive Science Society.*

Pylyshyn, Z.W. (1984) *Computation and cognition: Toward a foundation for cognitive science.* MIT Press/Bradford Books.

Riley, M.S. & Smolensky, P. (1984) A parallel model of (sequential) problem solving. *Proceedings of the Sixth Annual Conference of the Cognitive Science Society.*

Rumelhart, D.E. (1975) Notes on a schema for stories. In: *Representation and understanding*, ed. D.G. Bobrow and A. Collins, pp. 211–236. Academic.

Rumelhart, D.E. (1980) Schemata: The building blocks of cognition. In: *Theoretical issues in reading comprehension*, ed. R. Spiro, B. Bruce, and W. Brewer. Erlbaum.

Rumelhart, D.E., Hinton, G.E., & McClelland, J.L. (1986) Learning internal representations by error propagation. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Rumelhart, D.E. & McClelland, J.L. (1982) An interactive activation model of context effects in letter perception: Part 2. The contextual enhancement effect and some tests and extensions of the model. *Psychological Review* 89: 60–94.

Rumelhart, D.E. & McClelland, J.L. (1986) On learning the past tenses of English verbs. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Rumelhart, D.E., McClelland, J.L., & the PDP Research Group. (1986) *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*. MIT Press/Bradford Books.

Rumelhart, D.E., Smolensky, P., McClelland, J.L., and Hinton, G.E. (1986) Schemata and sequential thought processes in parallel distributed processing models. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Sejnowski, T.J. (1976) On the stochastic dynamics of neuronal interactions. *Biological cybernetics* 22: 203–211.

Sejnowski, T.J. & Rosenberg, C.R. (1986) NETtalk: A parallel network that learns to read aloud. Technical Report JHU/EECS-86/01, Department of Electrical Engineering and Computer Science, The Johns Hopkins University.

Shastri, L. (1985) Evidential reasoning in semantic networks: A formal theory and its parallel implementation. Technical Report TR 166, Department of Computer Science, University of Rochester.

Smolensky, P. (1983) Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence.*

Smolensky, P. (1984a) Harmony theory: thermal parallel models in a computational context. In: "Harmony theory: Problem solving, parallel cognitive models, and thermal physics," P. Smolensky and M. S. Riley. Technical Report 8404, Institute for Cognitive Science, University of California at San Diego.

Smolensky, P. (1984b) The mathematical role of self-consistency in parallel computation. *Proceedings of the Sixth Annual Conference of the Cognitive Science Society.*

Smolensky, P. (1986a) Information processing in dynamical systems: Foundations of harmony theory. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Smolensky, P. (1986b) Neural and conceptual interpretations of parallel distributed processing models. In: *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and*

biological models, J. L. McClelland, D. E. Rumelhart, & the PDP Research Group. MIT Press/Bradford Books.

Smolensky, P. (1986c) Formal modeling of subsymbolic processes: An introduction to harmony theory. In: *Directions in the science of cognition*, ed. N. E. Sharkey. Ellis Horwood.

Streater, R. & Wightman, A. S. (1964) *PCT, spin and statistics, and all that.* Benjamin.

Touretzky, D.S. & Hinton, G.E. (1985) Symbols among the neurons: Details of a connectionist inference architecture. *Proceedings of the International Joint Conference on Artificial Intelligence*.

Waltz, D.L. (1978) An English language question answering system for a large relational database. *Communications of the Association for Computing Machinery* 21: 526–539.

Waltz, D.L. & Pollock, J.B. (1985) Massively parallel parsing: A strongly interactive model of natural language interpretation. *Cognitive Science* 9: 51–74.

Table 1: Relations between the neural and subsymbolic architectures.

Cerebral Cortex		Connectionist Dynamical Systems
state defined by continuous numerical variables (potentials, synaptic areas, ...)	+	state defined by continuous numerical variables (activations, connection strengths)
state variables change continuously in time	+	state variables change continuously in time
inter-neuron interaction parameters changeable; seat of knowledge	+	inter-unit interaction parameters changeable; seat of knowledge
huge number of state variables	+	large number of state variables
high interactional complexity (highly nonhomogeneous interactions)	+	high interactional complexity (highly nonhomogeneous interactions)
neurons located in 2+1-d space have dense connectivity to nearby neurons have geometrically mapped connectivity to distant neurons	– – –	units have no spatial location uniformly dense connections
distal projections between areas have intricate topology	–	distal projections between node pools have simple topology
distal interactions mediated by discrete signals	–	all interactions non-discrete
intricate signal integration at single neuron	–	signal integration is linear
numerous signal types	–	single signal type
connections of a single neuron are all either excitatory or inhibitory	–	connection signs unconstrained

Table 2: Three cognitive systems and three levels of description

level *(process)*	**cognitive system**		
	brain	**subsymbolic**	**symbolic**
conceptual *(intuition)*	?	rough approximation ~	exact
conceptual *(conscious rule application)*	?	good approximation ≈	exact
subconceptual	good approximation ≈	exact	
neural	exact		