CHAPTER 6

Information Processing in Dynamical Systems: Foundations of Harmony Theory

P. SMOLENSKY

INTRODUCTION

The Theory of Information Processing

At this early stage in the development of cognitive science, methodological issues are both open and central. There may have been times when developments in neuroscience, artificial intelligence, or cognitive psychology seduced researchers into believing that their discipline was on the verge of discovering the secret of intelligence. But a humbling history of hopes disappointed has produced the realization that understanding the mind will challenge the power of all these methodologies combined.

The work reported in this chapter rests on the conviction that a methodology that has a crucial role to play in the development of cognitive science is *mathematical analysis*. The success of cognitive science, like that of many other sciences, will, I believe, depend upon the construction of a solid body of theoretical results: results that express in a mathematical language the conceptual insights of the field; results that squeeze all possible implications out of those insights by exploiting powerful mathematical techniques.

This body of results, which I will call the *theory of information processing*, exists because information is a concept that lends itself to mathematical formalization. One part of the theory of information processing is already well-developed. The classical theory of computation provides powerful and elegant results about the notion of *effective*

procedure, including languages for precisely expressing them and theoretical machines for realizing them. This body of theory grew out of mathematical logic, and in turn contributed to computer science, physical computing systems, and the theoretical paradigm in cognitive science often called *the (von Neumann) computer metaphor.*[1]

In his paper "Physical Symbol Systems," Allen Newell (1980) articulated the role of the mathematical theory of symbolic computation in cognitive science and furnished a manifesto for what I will call *the symbolic paradigm.* The present book offers an alternative paradigm for cognitive science, the *subsymbolic paradigm*, in which the most powerful level of description of cognitive systems is hypothesized to be lower than the level that is naturally described by symbol manipulation.

The fundamental insights into cognition explored by the subsymbolic paradigm do not involve effective procedures and symbol manipulation. Instead they involve the "spread of activation," relaxation, and statistical correlation. The mathematical language in which these concepts are naturally expressed are probability theory and the theory of dynamical systems. By dynamical systems theory I mean the study of sets of numerical variables (e.g., activation levels) that evolve in time in parallel and interact through differential equations. The classical theory of dynamical systems includes the study of natural physical systems (e.g., mathematical physics) and artificially designed systems (e.g., control theory). Mathematical characterizations of dynamical systems that formalize the insights of the subsymbolic paradigm would be most helpful in developing the paradigm.

This chapter introduces *harmony theory*, a mathematical framework for studying a class of dynamical systems that perform cognitive tasks according to the account of the subsymbolic paradigm. These dynamical systems can serve as models of human cognition or as designs for artificial cognitive systems. The ultimate goal of the enterprise is to develop a body of mathematical results for the theory of information processing that complements the results of the classical theory of (symbolic) computation. These results would serve as the basis for a manifesto for the subsymbolic paradigm comparable to Newell's manifesto for the symbolic paradigm. The promise offered by this goal will, I hope, be suggested by the results of this chapter, despite their very limited scope.

---

[1] Mathematical logic has recently given rise to another approach to formalizing information: *situation semantics* (Barwise & Perry, 1983). This is related to Shannon's (1948/1963) measure of information through the work of Dretske (1981). The approach of this chapter is more faithful to the probabilistic formulation of Shannon than is the symbolic approach of situation semantics. (This results from Dretske's move of identifying information with conditional probabilities of 1.)

It should be noted that harmony theory is a "theory" in the *mathematical* sense, not the *scientific* sense. By a "mathematical theory"—e.g., number theory, group theory, probability theory, the theory of computation—I mean a body of knowledge about a part of the ideal mathematical world; a set of definitions, axioms, theorems, and analytic techniques that are tightly interrelated. Such mathematical theories are distinct from scientific theories, which are of course bodies of knowledge about a part of the "real" world. Mathematical theories provide a language for expressing scientific theories; a given mathematical theory can be used to express a large class of scientific theories. Group theory, for example, provides a language for expressing many competing theories of elementary particles. Similarly, harmony theory can be used to express many alternative theories about various cognitive phenomena. The point is that without the concepts and techniques of the mathematical language of group theory, the formulation of *any* of the current scientific theories of elementary particles would be essentially impossible.

The goal of harmony theory is to provide a powerful language for expressing cognitive theories in the subsymbolic paradigm, a language that complements the existing languages for symbol manipulation. Since harmony theory is conceived as a language for using the subsymbolic paradigm to describe cognition, it embodies the fundamental scientific claims of that paradigm. But on many important issues, such as how knowledge is represented in detail for particular cases, harmony theory does not itself make commitments. Rather, it provides a language for stating alternative hypotheses and techniques for studying their consequences.

A Top-Down Theoretical Strategy

How can mathematical analysis be used to study the processing mechanisms underlying the performance of some cognitive task?

One strategy, often associated with David Marr (1982), is to characterize the task in a way that allows mathematical *derivation* of mechanisms that perform it. This *top-down* theoretical strategy is pursued in harmony theory. My claim is not that the strategy leads to descriptions that are *necessarily* applicable to all cognitive systems, but rather that the strategy leads to new insights, mathematical results, computer architectures, and computer models that fill in the relatively unexplored conceptual world of parallel, massively distributed systems that perform cognitive tasks. Filling in this conceptual world is a necessary subtask, I believe, for understanding how brains and minds are capable of intelligence and for assessing whether computers with novel architectures might share this capability.

The Centrality of Perceptual Processing

The cognitive task I will study in this chapter is an abstraction of the task of perception. This abstraction includes many cognitive tasks that are customarily regarded as much "higher level" than perception (e.g., intuiting answers to physics problems). A few comments on the role of perceptual processing in the subsymbolic paradigm are useful at this point.

The vast majority of cognitive processing lies between the highest cognitive levels of explicit logical reasoning and the lowest levels of sensory processing. Descriptions of processing at the extremes are relatively well-informed—on the high end by formal logic and on the low end by natural science. In the middle lies a conceptual abyss. How are we to conceptualize cognitive processing in this abyss?

The strategy of the symbolic paradigm is to conceptualize processing in the intermediate levels as symbol manipulation. Other kinds of processing are viewed as limited to extremely low levels of sensory and motor processing. Thus symbolic theorists climb *down* into the abyss, clutching a rope of symbolic logic anchored at the top, hoping it will stretch all the way to the bottom of the abyss.

The subsymbolic paradigm takes the opposite view, that intermediate processing mechanisms are of the same kind as perceptual processing mechanisms. Logic and symbol manipulation are viewed as appropriate descriptions only of the few cognitive processes that explicitly involve logical reasoning. Subsymbolic theorists climb *up* into the abyss on a perceptual ladder anchored at the bottom, hoping it will extend all the way to the top of the abyss.[2]

---

[2] There is no contradiction between working from lower level, perceptual processes up towards higher processes, and pursuing a top-down theoretical strategy. It is important to distinguish levels of *processing entities* from levels of *theoretical entities*. Higher level *processes* involve *computational* entities that are computationally distant from the peripheral, sensorimotor entities that comprise the "lowest level" of processing. These processing levels *taken together* form the processing system as a whole; they causally interact with each other through bottom-up and top-down *processing*. Higher level *theories* involve *descriptive* entities that are descriptively distant from entities that are directly part of an actual processing mechanism; these comprise the "lowest level" description. Each theoretical level *individually* describes the processing system as a whole; the interaction of descriptive levels is not *causal*, but *definitional*. (For example, changes in individual neural firing rates at the retina *cause* changes in individual firing rates in visual cortex after a delay related to causal information propagation. The same changes in individual retinal neuron firing rates *by definition* change the *average firing rates of pools* of retinal neurons; these higher level descriptive entities change instantly, without any causal information propagation from the lower level description.) Thus in harmony theory, models of higher level *processes* are derived from models of lower level, perceptual, processes, while lower level *descriptions* of these models are derived from higher level descriptions.

In this chapter, I will analyze an abstraction of the task of perception that encompasses many tasks, from low, through intermediate, to high cognitive levels. The analysis leads to a general kind of "perceptual" processing mechanism that is a powerful potential component of an information processing system. The abstract task I analyze captures a common part of the tasks of passing from an intensity pattern to a set of objects in three-dimensional space, from a sound pattern to a sequence of words, from a sequence of words to a semantic description, from a set of patient symptoms to a set of disease states, from a set of givens in a physics problem to a set of unknowns. Each of these processes is viewed as *completing an internal representation of a static state of an external world.* By suitably abstracting the task of interpreting a static *sensory* input, we can arrive at a theory of interpretation of static input *generally*, a theory of the *completion task* that applies to many cognitive phenomena in the gulf between perception and logical reasoning. An application that will be described in some detail is qualitative problem solving in circuit analysis.[3]

The central idea of the top-down theoretical strategy is that properties of the task are powerfully constraining on mechanisms. This idea can be well exploited within a perceptual approach to cognition, where the constraints on the perceptual task are characterized through the constraints operative in the external environment from which the inputs come. This permits an analysis of how internal representation of these constraints within the cognitive system itself allows it to perform its task. These kinds of considerations have been emphasized in the psychological literature prominently by Gibson and Shepard (see Shepard, 1984); they are fundamental to harmony theory.

Structure of the Chapter

The goal of harmony theory is to develop a mathematical theory of information processing in the subsymbolic paradigm. However, the theory grows out of ideas that can be stated with little or no mathematics. The organization of this chapter reflects an attempt to ensure that the central concepts are not obscured by mathematical opacity. The analysis will be presented in three parts, each part increasing in the level of formality and detail. My hope is that the slight redundancy

[3] Many cognitive tasks involve interpreting or controlling events that unfold over an extended period of time. To deal properly with such tasks, harmony theory must be extended from the interpretation of *static* environments to the interpretation of *dynamic* environments.

introduced by this expository organization will be repaid by greater accessibility.

Section 1 is a top-down presentation of how the perceptual perspective on cognition leads to the basic features of harmony theory. This presentation starts with a particular perceptual model, the letter-perception model of McClelland and Rumelhart (1981), and abstracts from it general features that can apply to modeling of higher cognitive processes. Crucial to the development is a particular formulation of aspects of schema theory, along the lines of Rumelhart (1980).

Section 2, the majority of the chapter, is a bottom-up presentation of harmony theory that starts with the primitives of the knowledge representation. Theorems are informally described that provide a competence theory for a cognitive system that performs the completion task, a machine that realizes this theory, and a learning procedure through which the machine can absorb the necessary information from its environment. Then an application of the general theory is described: a model of intuitive, qualitative problem-solving in elementary electric circuits. This model illustrates several points about the relation between symbolic and subsymbolic descriptions of cognitive phenomena; for example, it furnishes a sharp contrast between the description at these two levels of the nature and acquisition of expertise.

The final part of the chapter is an Appendix containing a concise but self-contained formal presentation of the definitions and theorems.

SECTION 1: SCHEMA THEORY AND SELF-CONSISTENCY

THE LOGICAL STRUCTURE OF HARMONY THEORY

The logical structure of harmony theory is shown schematically in Figure 1. The box labeled *Mathematical Theory* represents the use of mathematical analysis and computer simulation for drawing out the implications of the fundamental principles. These principles comprise a mathematical characterization of computational requirements of a cognitive system that performs the completion task. From these principles

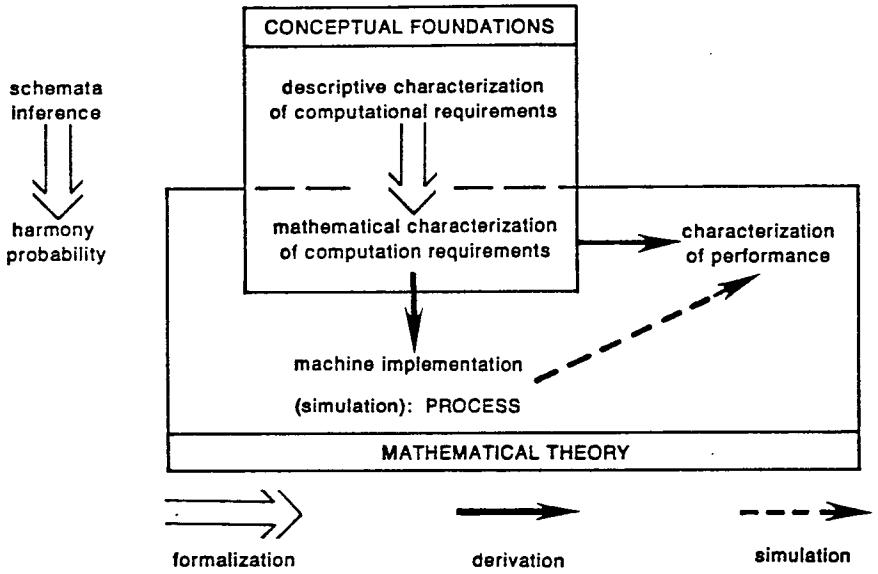

FIGURE 1. The logical structure of harmony theory.

it is possible to mathematically analyze aspects of the resulting performance as well as rigorously *derive* the rules for a machine implementing the computational requirements. The rules defining this machine have a different status from those defining most other computer models of cognition: They are not ad hoc, or post hoc; rather they are logically derived from a set of computational requirements. This is one sense in which harmony theory has a top-down theoretical development.

Where do the "mathematically characterized computational requirements" of Figure 1 come from? They are a formalization of a descriptive characterization of cognitive processing, a simple form of *schema theory*. In Section 1 of this chapter, I will give a description of this form of schema theory and show how to transform the descriptive characterization into a mathematical one—how to get from the *conceptual* box of Figure 1 into the *mathematical* box. Once we are in the formal world, mathematical analysis and computer simulation can be put to work.

Throughout Section 1, the main points of the development will be explicitly enumerated.

Point 1. The mathematics of harmony theory is founded on familiar concepts of cognitive science: inference through activation of schemata.

DYNAMIC CONSTRUCTION OF SCHEMATA

The basic problem can be posed à la Schank (1980). While eating at a fancy restaurant, you get a headache. Without effort, you ask the waitress if she could possibly get you an aspirin. How is this plan created? You have never had a headache in a restaurant before. Ordinarily, when you get a headache your plan is to go to your medicine cabinet and get yourself some aspirin. In the current situation, this plan must be modified by the knowledge that in good restaurants, the management is willing to expend effort to please its customers, and that the waitress is a liaison to that management.

The cognitive demands of this situation are schematically illustrated in Figure 2. Ordinarily, the restaurant context calls for a "restaurant script" which supports the planning and inferencing required to reach the usual goal of getting a meal. Ordinarily, the headache context calls for a "headache script" which supports the planning required to get aspirin in the usual context of home. The completely novel context of a headache in a restaurant calls for a special-purpose script integrating the knowledge that ordinarily manifests itself in two separate scripts.

What kind of cognitive system is capable of this degree of flexibility? Suppose that the knowledge base of the system does *not* consist of a set of scripts like the restaurant script and the headache script. Suppose

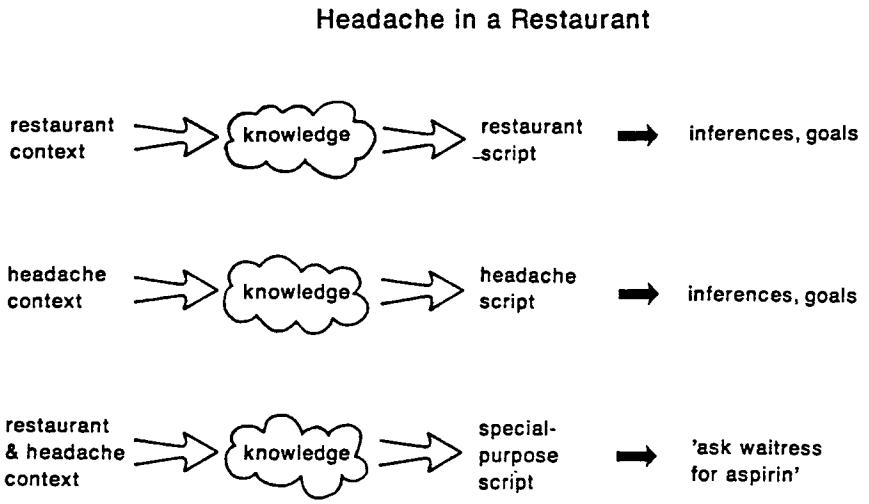

FIGURE 2. In three different contexts, the knowledge base must produce three different scripts.

instead that the knowledge base is a set of *knowledge atoms* that configure themselves dynamically in each context to form tailor-made scripts. This is the fundamental idea formalized in harmony theory.[4]

The degree of flexibility demanded of scripts is equaled by that demanded of all conceptual structures.[5] For example, metaphor is an extreme example of the flexibility demanded of word meanings; even so-called literal meaning on closer inspection actually relies on extreme flexibility of knowledge application (Rumelhart, 1979). In this chapter I will consider knowledge structures that embody our knowledge of objects, words, and other concepts of comparable complexity; these I will refer to as *schemata*. The defining properties of schemata are that they have conceptual interpretations and that they *support inference*.

For lack of a better term, I will use *knowledge atoms* to refer to the elementary constituents of which I assume schemata to be composed.[6] These atoms will shortly be given a precise description; they will be interpreted as a particular instantiation of the idea of *memory trace*.

Point 2. At the time of inference, stored knowledge atoms are dynamically assembled into context-sensitive schemata.

This view of schemata was explicitly articulated in Feldman (1981). It is in part embodied in the McClelland and Rumelhart (1981) letter-perception model (see Chapter 1). One of the observed phenomena accounted for by this model is the facilitation of the perception of letters that are embedded in words. Viewing the perception of a letter as the result of a perceptual inference process, we can say that this inference is supported by a *word schema* that appears in the model as a single processing unit that encodes the knowledge of the spelling of that word. This is *not* an instantiation of the view of schemata as dynamically created entities.

---

[4] Schank (1980) describes a *symbolic* implementation of the idea of dynamic script construction; harmony theory constitutes a *subsymbolic* formalization.

[5] Hofstadter has long been making the case for the inadequacy of traditional symbolic descriptions to cope with the power and flexibility of concepts. For his most recent argument, see Hofstadter (1985). He argues for the need to admit the approximate nature of symbolic descriptions, and to explicitly consider processes that are *subcognitive*. In Hofstadter (1979, p. 324ff), this same case was phrased in terms of the need for "active symbols," of which the "schemata" described here can be viewed as instances.

[6] A physicist might call these particles *gnosons* or *sophons*, but these terms seem quite uneuphonious. An acronym for *Units for Constructing Schemata Dynamically* might serve, but would perhaps be taken as an advertising gimmick. So I have stuck with "knowledge atoms."

However, the model also accounts for the observed facilitation of letter perception within orthographically regular nonwords or *pseudowords* like *MAVE*. When the model processes this stimulus, several word units become and stay quite active, including *MAKE, WAVE, HAVE*, and other words orthographically similar to *MAVE*. In this case, the perception of a letter in the stimulus is the result of an inference process that is supported by the *collection* of activated units. This collection is a *dynamically created pseudoword schema*.

When an orthographically irregular nonword is processed by the model, letter perception is slowest. As in the case of pseudowords, many word units become active. However, none become very active, and very many are equally active, and these words have very little similarity to each other, so they do not support inference about the letters effectively. Thus the knowledge base is incapable of creating schemata for irregular nonwords.

Point 3. Schemata are coherent assemblies of knowledge atoms; only these can support inference.

Note that schemata are created *simply by activating the appropriate atoms*. This brings us to what was labeled in Figure 1 the "descriptively characterized computational requirements" for harmony theory:

Point 4: The harmony principle. The cognitive system is an engine for activating coherent assemblies of atoms and drawing inferences that are consistent with the knowledge represented by the activated atoms.

Subassemblies of activated atoms that tend to recur exactly or approximately are the schemata.

This principle focuses attention on the notion of *coherency* or *consistency*. This concept will be formalized under the name of *harmony*, and its centrality is acknowledged by the name of the theory.

MICRO- AND MACROLEVELS

It is important to realize that harmony theory, like all subsymbolic accounts of cognition, exists on two distinct levels of description: a microlevel involving knowledge atoms and a macrolevel involving schemata (see Chapter 14). These levels of description are completely analogous to other micro- and macrotheories, for example, in physics. The microtheory, quantum physics, is assumed to be universally valid. Part of its job as a theory is to explain why the approximate macrotheory, classical physics, works when it does and why it breaks

down when it does. Understanding of physics requires understanding *both* levels of theory *and* the relation between them.

In the subsymbolic paradigm in cognitive science, it is equally important to understand the two levels and their relationship. In harmony theory, the microtheory prescribes the nature of the atoms, their interaction, and their development through experience. This description is assumed to be a universally valid description of cognition. It is also assumed (although this has yet to be explicitly worked out) that in performing certain cognitive tasks (e.g., logical reasoning), a higher level description is a valid approximation. This macrotheory describes schemata, their interaction, and their development through experience.

One of the features of the formalism of harmony theory that distinguishes it from most subsymbolic accounts of cognition is that it exploits a formal isomorphism with statistical physics. Since the main goal of statistical physics is to relate the microscopic description of matter to its macroscopic properties, harmony theory can bring the power of statistical physics concepts and techniques to bear on the problem of understanding the relation between the micro- and macroaccounts of cognition.

THE NATURE OF KNOWLEDGE

In the previous section, the letter-perception model was used to illustrate the dynamic construction of schemata from constituent atoms. However, it is only pseudowords that correspond to composite schemata; word schemata are single atoms. We can also represent words as composite schemata by using digraph units at the upper level instead of four-letter word units. A portion of this modified letter-perception model is shown in Figure 3. Now the processing of a four-letter word involves the activation of a set of digraph units, which are the knowledge atoms of this model. Omitted from the figure are the

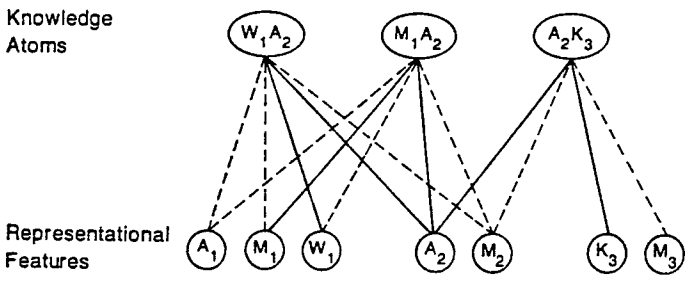

FIGURE 3. A portion of a modified reading model.

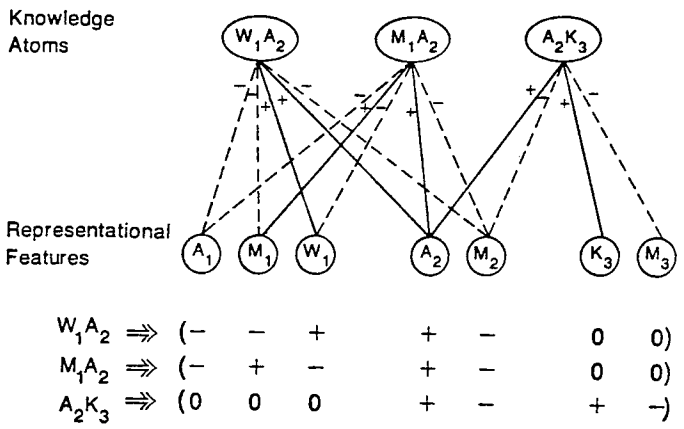

FIGURE 4. Each knowledge atom is a vector of +, −, and 0 values of the representational feature nodes.

line-segment units, which are like those in the original letter-perception model.

This simple model illustrates several points about the nature of knowledge atoms in harmony theory. The digraph unit W_1A_2 represents a pattern of values over the letter units: W_1 and A_2 on, with all other letter units for positions 1 and 2 off. This pattern is shown in Figure 4, using the labels +, −, and 0 to denote *on*, *off*, and *irrelevant*. These indicate whether there is an excitatory connection, inhibitory connection, or no connection between the corresponding nodes.[7]

Figure 4 shows the basic structure of harmony models. There are atoms of knowledge, represented by nodes in an upper layer, and a lower layer of nodes that comprises a representation of the state of the perceptual or problem domain with which the system deals. Each node is a *feature* in the representation of the domain. We can now view "atoms of knowledge" like W_1 and A_2 in several ways. Mathematically, each atom is simply a *vector* of +, −, and 0 values, one for each node in the lower, representation layer. This pattern can also be viewed as a *fragment* of a percept: The 0 values mark those features omitted in the fragment. This fragment can in turn be interpreted as a *trace* left behind in memory by perceptual experience.

[7] Omitted are the knowledge atoms that relate the letter nodes to the line segment nodes. Both line segment and letter nodes are in the lower layer, and all knowledge atoms are in the upper layer. Hierarchies in harmony theory are imbedded within an architecture of only two layers of nodes, as will be discussed in Section 2.

Point 5. Knowledge atoms are fragments of representations that accumulate with experience.

THE COMPLETION TASK

Having specified more precisely what the atoms of knowledge are, it is time to specify the task in which they are used.

Many cognitive tasks can be viewed as inference tasks. In problem solving, the role of inference is obvious; in perception and language comprehension, inference is less obvious but just as central. In harmony theory, a tightly prescribed but extremely general inferential task is studied: the *completion task*. In a problem-solving completion task, a partial description of a situation is given (for example, the initial state of a system); the problem is to complete the description to fill in the missing information (the final state, say). In a story understanding completion task, a partial description of some events and actors' goals is given; comprehension involves filling in the missing events and goals. In perception, the stimulus gives values for certain low-level features of the environmental state, and the perceptual system must fill in values for other features. In general, in the completion task some features of an environmental state are given as input, and the cognitive system must complete that input by assigning likely values to unspecified features.

A simple example of a completion task (Lindsay & Norman, 1972) is shown in Figure 5. The task is to fill in the features of the obscured portions of the stimulus and to decide what letters are present. This task can be performed by the model shown in Figure 3, as follows. The stimulus assigns values of *on* and *off* to the unobscured letter features. What happens is summarized in Table 1.

Note that which atoms are activated affects how the representation is

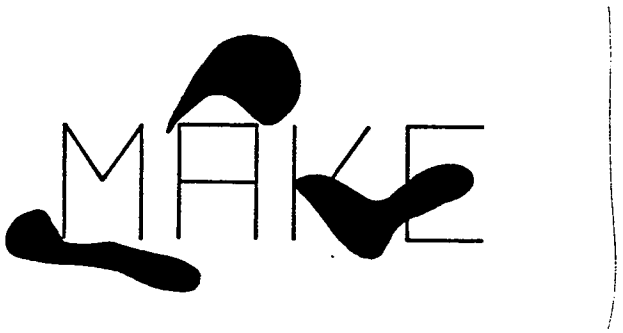

FIGURE 5. A perceptual completion task.

TABLE I

A PROCEDURE FOR PERFORMING THE COMPLETION TASK

Input:	Assign values to some features in the representation
Activation:	Activate atoms that are *consistent* with the representation
Inference:	Assign values to unknown features of representation that are *consistent* with the active knowledge

filled in, and how the representation is filled in affects which atoms are activated. The activation and inference processes mutually constrain each other; these processes must run in parallel. Note also that all the decisions come out of a striving for *consistency*.

Point 6. Assembly of schemata (activation of atoms) and inference (completing missing parts of the representation) are both achieved by finding maximally self-consistent states of the system that are also consistent with the input.

The completion of the stimulus shown in Figure 5 is shown in Figure 6. The consistency is high because wherever an active atom is

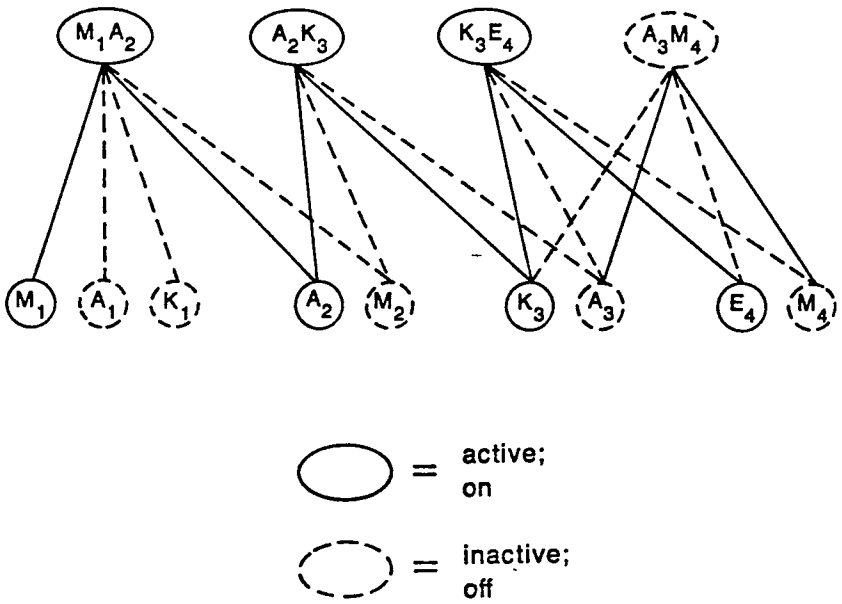

FIGURE 6. The state of the network in the completion of the stimulus shown in Figure 5.

connected to a representational feature by a + (respectively, −) connection, that feature has value *on* (respectively, *off*). In fact, we can define a very simple measure of the degree of self-consistency just by considering all active atoms, counting +1 for every agreement between one of its connections and the value of the corresponding feature, and counting −1 for every disagreement. (Here + with *on* or − with *off* constitutes agreement.) This is the simplest example of a *harmony function*—and brings us into the mathematical formulation.

THE HARMONY FUNCTION

Point 6 asserts that a central cognitive process is the construction of cognitive states that are "maximally self-consistent." To make this precise, we need only measure that self-consistency.

Point 7. The self-consistency of a possible state of the cognitive system can be assigned a quantitative value by a harmony function, H.

Figure 7 displays a harmony function that generalizes the simple example discussed in the preceding paragraph. A state of the system is defined by a set of atoms which are *active* and a vector of values for all representational features. The harmony of such a state is the sum of terms, one for each active atom, weighted by the *strength* of that atom. Each weight multiplies the self-consistency between that particular atom and the vector of representational feature values. That self-consistency is the similarity between the vector of features defining the atom (the vector of its connections) and the representational feature vector. In the simplest case discussed above, the function h that measures this similarity is just the number of agreements between these vectors minus the number of disagreements. For reasons to be discussed, I have used a slightly more complicated version of h in which the simpler form is first divided by the number of (nonzero) connections to the atom, and then a fixed value κ is subtracted.

$$\text{harmony}_{\substack{\text{knowledge}\\\text{base}}}\left(\text{representational feature vector, activations}\right) =$$

$$\sum_{\substack{\text{atoms}\\\alpha}} \left(\begin{matrix}\text{strength of}\\\text{atom } \alpha\end{matrix}\right)\left(\begin{matrix}0 \text{ if atom}\\\alpha \text{ inactive;}\\1 \text{ if active}\end{matrix}\right) \text{similarity}\left(\begin{matrix}\text{feature vector}\\\text{of atom } \alpha\end{matrix}, \begin{matrix}\text{representational}\\\text{feature vector}\end{matrix}\right)$$

FIGURE 7. A schematic representation for a harmony function.

A PROBABILISTIC FORMULATION OF SCHEMA THEORY

The next step in the theoretical development requires returning to the higher level, symbolic description of inference, and to a more detailed discussion of schemata.

Consider a typical inference process described with schemata. A child is reading a story about presents, party hats, and a cake with candles. When asked questions, the child says that the girl getting the presents is having a birthday. In the terminology of schema theory, while reading the story, the child's *birthday party schema* becomes active and allows many inferences to be made, filling in details of the scene that were not made explicit in the story.

The birthday party schema is presumed to be a knowledge structure that contains *variables* like *birthday cake, guest of honor, other guests, gifts, location*, and so forth. The schema contains information on how to assign values to these variables. For example, the schema may specify: *default values* to be assigned to variables in the absence of any counterindicating information; *value restrictions* limiting the kind of values that can be assigned to variables; and *dependency* information, specifying how assigning a particular value to one variable affects the values that can be assigned to another variable.

A convenient framework for concisely and uniformly expressing all this information is given by *probability theory*. The default value for a variable can be viewed as its most probable value: the mode of the marginal probability distribution for that variable. The value restrictions on a variable specify the values for which it has nonzero probability: the support of its marginal distribution. The dependencies between variables are expressed by their statistical correlations, or, more completely, by their joint probability distributions.

So the birthday party schema can be viewed as containing information about the probabilities that its variables will have various possible values. These are clearly statistical properties of the particular domain or *environment* in which the inference task is being carried out. In reading the story, the child is given a partial description of a scene from the everyday environment—the values of some of the features used to represent that scene—and to understand the story, the child must *complete* the description by filling in the values for the unknown features. These values are assigned in such a way that the resulting scene has the highest possible probability. The birthday party schema contains the probabilistic information needed to carry out these inferences.

In a typical cognitive task, many schemata become active at once and interact heavily during the inference process. Each schema contains probabilistic information for its own variables, which are only a fraction

of the complete set of variables involved in the task. To perform a completion, the most probable set of values must be assigned to the unknown variables, using the information in all the active schemata.

This probabilistic formulation of these aspects of schema theory can be simply summarized as follows.

Point 8. Each schema encodes the statistical relations among a few representational features. During inference, the probabilistic information in many active schemata are dynamically folded together to find the most probable state of the environment.

Thus the statistical knowledge encoded in all the schemata allow the estimation of the relative probabilities of possible states of the environment. How can this be done?

At the macrolevel of schemata and variables, coordinating the folding together of the information of many schemata is difficult to describe. The inability to devise procedures that capture the flexibility displayed in human use of schemata was in fact one of the primary historical reasons for turning to the microlevel description (see Chapter 1). We therefore return to the microdescription to address this difficult problem.

At the microlevel, the probabilistic knowledge in the birthday party schema is distributed over many knowledge atoms, each carrying a small bit of statistical information. Because these atoms all tend to match the representation of a birthday party scene, they can become active together; in some approximation, they tend to function collectively, and in that sense they comprise a schema. Now, when many schemata are active at once, that means the knowledge atoms that comprise them are simultaneously active. At the microlevel, there is no real difference between the decisions required to activate the appropriate atoms to instantiate many schemata simultaneously and the decisions required to activate the atoms to instantiate a single schema. A computational system that can dynamically create a schema when it is needed can also dynamically create many schemata when they are needed. When atoms, not schemata, are the elements of computation, the problem of coordinating many schemata becomes subsumed in the problem of activating the appropriate atoms. And this is the problem that the harmony function, the measure of self-consistency, was created to solve.

HARMONY THEORY

According to Points 2, 6, and 7, schemata are collections of knowledge atoms that become active in order to maximize harmony,

and inferences are also drawn to maximize harmony. This suggests that the probability of a possible state of the environment is estimated by computing its harmony: the higher the harmony, the greater the probability. In fact, from the mathematical properties of probability and harmony, in Section 2 we will show the following:

Point 9. The relationship between the harmony function H and estimated probabilities is of the form

$$\text{probability} \propto e^{H/T}$$

where T is some constant that cannot be determined a priori.

This relationship between probability and harmony is mathematically identical to the relationship between probability and (minus) energy in statistical physics: the Gibbs or Boltzmann law. This is the basis of the isomorphism between cognition and physics exploited by harmony theory. In statistical physics, H is called the *Hamiltonian function*; it measures the energy of a state of a physical system. In physics, T is the *temperature* of the system. In harmony theory, T is called the *computational temperature* of the cognitive system. When the temperature is very high, completions with high harmony are assigned estimated probabilities that are only slightly higher than those assigned to low harmony completions; the environment is treated as *more random* in the sense that all completions are estimated to have roughly equal probability. When the temperature is very low, only the completions with highest harmony are given nonnegligible estimated probabilities.[8]

Point 10. The lower the computational temperature, the more the estimated probabilities are weighted towards the completions of highest harmony.

In particular, the very best completion can be found by lowering the temperature to zero. This process, *cooling*, is fundamental to harmony theory. Concepts and techniques from thermal physics can be used to understand and analyze decision-making processes in harmony theory.

A technique for performing Monte Carlo computer studies of thermal systems can be readily adapted to harmony theory.

Point 11. A massively parallel stochastic machine can be designed that performs completions in accordance with Points 1-10.

---

[8] Since harmony corresponds to *minus* energy, at low physical temperatures only the state with the *lowest* energy (the *ground state*) has nonnegligible probability.

For a given harmony model (e.g., that of Figure 4), this machine is constructed as follows. Every node in the network becomes a simple processor, and every link in the network becomes a communication link between two processors. The processors each have two possible values (+1 and −1 for the representational feature processors; 1 = *active* and 0 = *inactive* for the knowledge atom processors). The input to a completion problem is provided by fixing the values of some of the feature processors. Each of the other processors continually updates its value by making stochastic decisions based on the harmony associated at the current time with its two possible values. It is most likely to choose the value that corresponds to greater harmony; but with some probability—greater the higher is the computational temperature T—it will make the other choice. Each processor computes the harmony associated with its possible values by a numerical calculation that uses as input the numerical values of all the other processors to which it is connected. Alternately, all the atom processors update in parallel, and then all the feature processors update in parallel. The process repeats many times, implementing the procedure of Table 1. All the while, the temperature T is lowered to zero, pursuant to Point 10. It can be proved that the machine will eventually "freeze" into a completion that maximizes the harmony.

I call this machine *harmonium* because, like the Selfridge and Neisser (1960) pattern recognition system *pandemonium*, it is a parallel distributed processing system in which many atoms of knowledge are simultaneously "shouting" out their little contributions to the inference process; but unlike pandemonium, there is an explicit method to the madness: the collective search for maximal harmony.[9]

The final point concerns the account of learning in harmony theory.

> *Point 12. There is a procedure for accumulating knowledge atoms through exposure to the environment so that the system will perform the completion task optimally.*

The precise meaning of "optimality" will be an important topic in the subsequent discussion.

This completes the descriptive account of the foundations of harmony theory. Section 2 fills in many of the steps and details omitted

---

[9] Harmonium is closely related to the *Boltzmann machine* discussed in Chapter 7. The basic dynamics of the machines are the same, although there are differences in most details. In the Appendix, it is shown that in a certain sense the Boltzmann machine is a special case of harmonium, in which knowledge atoms connected to more than two features are forbidden. In another sense, harmonium is a special case of the Boltzmann machine, in which the connections are restricted to go only between two layers.

above, and reports the results of some particular studies. The most formal matters are treated in the Appendix.

SECTION 2: HARMONY THEORY

> *. . . the privileged unconscious phenomena, those susceptible of becoming conscious, are those which . . . affect most profoundly our emotional sensibility . . . Now, what are the mathematic entities to which we attribute this character of beauty and elegance . . . ? They are those whose elements are harmoniously disposed so that the mind without effort can embrace their totality while realizing the details. This harmony is at once a satisfaction of our esthetic needs and an aid to the mind, sustaining and guiding. . . . Figure the future elements of our combinations as something like the unhooked atoms of Epicurus. . . . They flash in every direction through the space . . . like the molecules of a gas in the kinematic theory of gases. Then their mutual impacts may produce new combinations.*
>
> Henri Poincaré (1913)
> Mathematical Creation[10]

In Section 1, a top-down analysis led from the demands of the completion task and a probabilistic formulation of schema theory to perceptual features, knowledge atoms, the central notion of harmony, and the role of harmony in estimating probabilities of environmental states. In Section 2, the presentation will be bottom-up, starting from the primitives.

KNOWLEDGE REPRESENTATION

Representation Vector

At the center of any harmony theoretic model of a particular cognitive process is a set of *representational features* $r_1, r_2, \cdots$. These

[10] I am indebted to Yves Chauvin for recently pointing out this remarkable passage by the great mathematician. See also Hofstadter (1985, pp. 655-656).

features constitute the cognitive system's representation of possible states of the environment with which it deals. In the environment of visual perception, these features might include pixels, edges, depths of surface elements, and identifications of objects. In medical diagnosis, features might be symptoms, outcomes of tests, diseases, prognoses, and treatments. In the domain of qualitative circuit analysis, the features might include *increase in current through resistor x* and *increase in voltage drop across resistor x*.

The representational features are variables that I will assume take on binary values that can be thought of as *present* and *absent* or *true* and *false*. Binary values contain a tremendous amount of representational power, so it is not a great sacrifice to accept the conceptual and technical simplification they afford. It will turn out to be convenient to denote *present* and *absent* respectively by $+1$ and -1, or, equivalently, $+$ and $-$. Other values could be used if corresponding modifications were made in the equations to follow. The use of continuous numerical feature variables, while introducing some additional technical complexity, would not affect the basic character of the theory.[11]

A *representational state* of the cognitive system is determined by a collection of values for all the representational variables $\{r_i\}$. This collection can be designated by a list or vector of $+$'s and $-$'s: the *representation vector* $\mathbf{r}$.

Where do the features used in the representation vector come from? Are they "innate" or do they develop with experience? These crucial questions will be deferred until the last section of this chapter. The evaluation of various possible representations for a given environment and the study of the development of good representations through exposure to the environment is harmony theory's *raison d'être*. But a prerequisite for understanding the appropriateness of a representation is understanding how the representation supports performance on the task for which it used; that is the primary concern of this chapter. For now, we simply assume that somehow a set of representational features has already been set up: by a programmer, or experience, or evolution.

---

[11] While continuous values make the *analysis* more complex, they may well improve the performance of the simulation models. In simulations with discrete values, the system state jumps between corners of a hypercube; with continuous values, the system state crawls smoothly around inside the hypercube. It was observed in the work reported in Chapter 14 that "bad" corners corresponding to stable nonoptimal completions (local harmony maxima) were typically *not* visited by the smoothly moving continuous state; these corners typically *are* visited by the jumping discrete state and can only be escaped from through thermal stochasticity. Thus continuous values may sometimes eliminate the need for stochastic simulation.

Activation Vector

The representational features serve as the blackboard on which the cognitive system carries out its computations. The *knowledge* that guides those computations is associated with the second set of entities, the *knowledge atoms*. Each such atom α is characterized by a *knowledge vector* $\mathbf{k}_\alpha$, which is a list of $+1$, -1, and 0 values, one for each representation variable r_i. This list encodes a piece of knowledge that specifies what value each r_i should have: $+1$, -1, or unspecified (0).

Associated with knowledge atom α is its *activation variable*, a_α. This variable will also be taken to be binary: 1 will denote active; 0, inactive. Because harmony theory is probabilistic, degrees of activation are represented by varying probability of being active rather than varying values for the activation variable. (Like continuous values for representation variables, continuous values for activation variables could be incorporated into the theory with little difficulty, but a need to do so has not yet arisen.) The list of $\{0,1\}$ values for the activations $\{a_\alpha\}$ comprises the *activation vector* $\mathbf{a}$.

Knowledge atoms encode subpatterns of feature values that occur in the environment. The different frequencies with which various such patterns occur is encoded in the set of *strengths*, $\{\sigma_\alpha\}$, of the atoms.

In the example of qualitative circuit analysis, each knowledge atom records a pattern of qualitative changes in some of the circuit features (currents, voltages, etc.). These patterns are the ones that are consistent with the laws of physics, which are the constraints characterizing the circuit environment. Knowledge of the laws of physics is encoded in the set of knowledge atoms. For example, the atom whose knowledge vector contains all zeroes except those features encoding the pattern $<$*current decreases, voltage decreases, resistance increases*$>$ is one of the atoms encoding qualitative knowledge of Ohm's law. Equally important is the *absence* of an atom like one encoding the pattern $<$*current increases, voltage decreases, resistance increases*$>$, which violates Ohm's law.

There is a very useful graphical representation for knowledge atoms; it was illustrated in Figure 4 and is repeated as Figure 8. The representational features are designated by nodes drawn in a lower layer; the activation variables are depicted by nodes drawn in an upper layer. The connections from an activation variable a_α to the representation variables $\{r_i\}$ show the knowledge vector $\mathbf{k}_\alpha$. When $\mathbf{k}_\alpha$ contains a $+$ or $-$ for r_i, the connection between a_α and r_i is labeled with the appropriate sign; when $\mathbf{k}_\alpha$ contains a 0 for r_i, the connection between a_α and r_i is omitted.

In Figure 8, all atoms are assumed to have unit strength. In general, different atoms will have different strengths; the strength of each atom

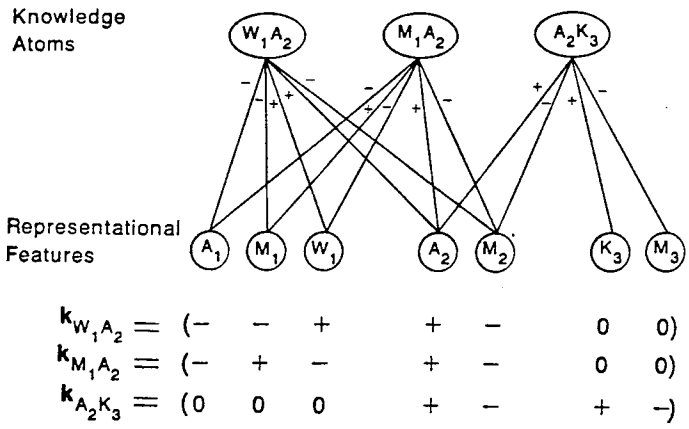

FIGURE 8. The graphical representation of a particular harmony model.

would them be indicated above the atom in the drawing. (For the completely general case, see Figure 13.)

Hierarchies and the Architecture of Harmony Networks

One of the characteristics that distinguishes harmony models from other parallel network models is that the graph *always* contains two layers of nodes, with rather different semantics. As in many networks, the nodes in the upper layer correspond to patterns of values in the lower layer. In the letter-perception model of McClelland and Rumelhart, for example, the word nodes correspond to patterns over the letter nodes, and the letter nodes in turn correspond to patterns over the line-segment nodes. The letter-perception model is typical in its *hierarchical structure*: The nodes are stratified into a sequence of several layers, with nodes in one layer being connected only to nodes in adjacent layers. Harmony models use only two layers.

The formalism could be extended to many layers, but the use of two layers has a principled foundation in the semantics of these layers. The nodes in the representation layer *support representations of the environment at all levels of abstractness*. In the case of written words, this layer could support representation at the levels of line segments, letters, and words, as shown schematically in Figure 9. The upper, knowledge, layer encodes the patterns among these representations. If information is given about line segments, then some of the knowledge atoms

connect that information with the letter nodes, completing the representation to include letter recognition. Other knowledge atoms connect patterns on the letter nodes with word nodes, and these complete the representation to include word recognition.

The pattern of connectivity of Figure 9 allows the network to be redrawn as shown in Figure 10. This network shows an alternation of representation and knowledge nodes, restoring the image of a series of layers. In this sense, "vertically" hierarchical networks of many layers can be imbedded as "horizontally" hierarchical networks within a two-layer harmony network.

Figure 10 graphically displays the fact that in a harmony architecture, the nodes that encode patterns are not part of the representation; there is a firm distinction between representation and knowledge nodes. This distinction is not made in the original letter-perception model, where the nodes that detect a pattern over the line-segment features are identical with the nodes that actually represent the presence of letters. This distinction seems artificial; why is it made?

I claim that the artificiality actually resides in the original letter-perception model, in which the presence of a letter can be identified with a single pattern over the primitive graphical features (line segments). In a less idealized reading task, the presence of a letter would have to be inferable from many different combinations of primitive graphical features. In harmony theory, the idea is that there would be a set of representation nodes dedicated to the representation of the presence of letters independent of their shapes, sizes, orientations, and so forth. There would also be a set of representation nodes for graphical

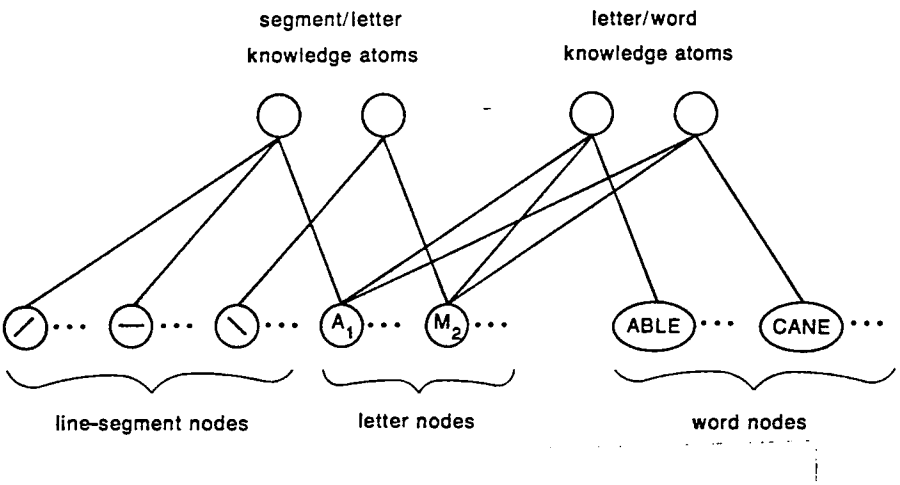

FIGURE 9. The representational features support representations at all levels of abstractness.

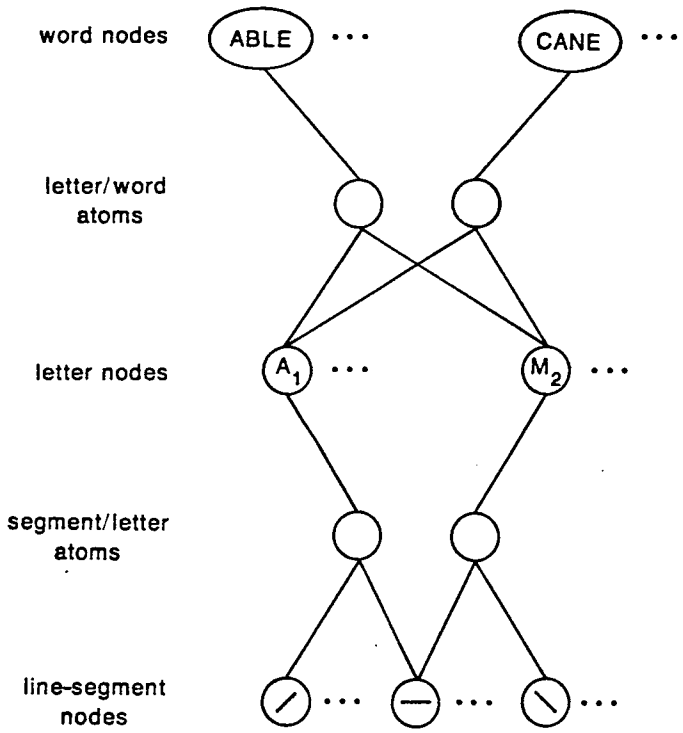

FIGURE 10. A rearrangement of the network of Figure 9.

features, and for each letter there would be *a multitude* of knowledge atoms, each relating a particular configuration of graphical features with the representation of that letter. Thus the knowledge or *schema* for that letter would be distributed over many knowledge atoms, all of which would be involved in setting up the same representation on the letter nodes. To provide a broader context, Figure 11 schematically depicts a possible model for language processing. The full representation consists of graphical features, phonological features, syntactic features, and semantic features. Some of the knowledge atoms provide connections among features within a single category, while others connect features in different categories. The nodes in the upper layer do not themselves comprise parts of the representation, but rather encode *relations between* parts of the representation.

The advantages of the two-layer scheme come from simplicity and uniformity. There are no connections within layers, only between layers. This simplifies mathematical analysis considerably and permits a

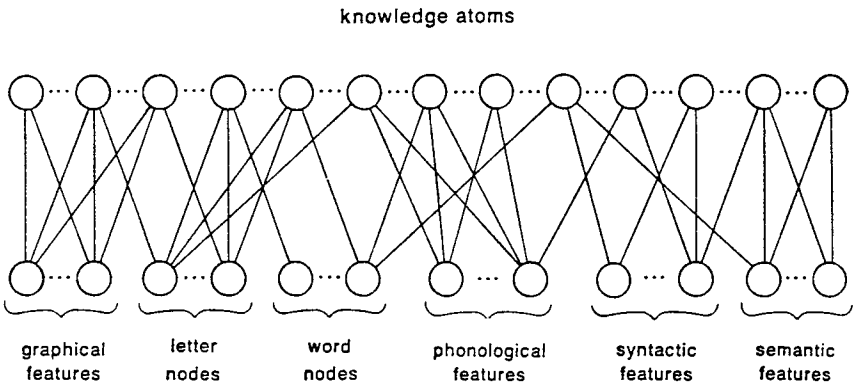

FIGURE 11. A complete model for language processing would involve representational variables of many types, and the atoms relating them.

truly parallel implementation. The uniformity means that we can imagine a system starting out with an "innate" two-layer structure and *learning* a pattern of connections like that of Figure 9, i.e., learning a hierarchical representation scheme that was in no sense put into the model in advance. The formalism is set up to analyze the environmental conditions under which certain kinds of representations (e.g., hierarchical ones) might emerge or be expedient.

The lack of within-layer connections in harmony networks is symptomatic of a major difference between the goals of harmony theory and the goals of other similar approaches. The effect of a binary connection between two representation nodes can be achieved by creating a pair of upper level nodes that connect to the two lower level nodes.[12] Thus we can dispense with lower level connections at the cost of creating upper level nodes. *Harmony theory has been developed with a systematic commitment to buy simplicity with extra upper level nodes.* The hope is that by placing all the knowledge in the patterns encoded by knowledge atoms, we will be better able to *understand* the function and structure of the models. This explains why restrictions have been placed on the network that to many would seem extraordinarily confining.

If the goal is instead to get the most "intelligent" performance out of the fewest number of nodes and connections, it is obviously wiser to

[12] A negative connection between two lower level nodes means that the value pairs (+,−) and (−,+) are favored relative to the other two pairs. This effect can be achieved by creating two knowledge atoms that each encode one of the two favored patterns. A positive connection similarly can be replaced by two atoms for the patterns (+,+) and (−,−).

allow arbitrary connectivity patterns, weights, and thresholds, as in the Boltzmann machine. There are, however, theoretical disadvantages to having so many degrees of freedom, both in psychological modeling and in artificial intelligence applications. Too many free parameters in a psychological model make it too theoretically unconstrained and therefore insufficiently instructive. And as suggested in Chapter 7, networks that take advantage of all these degrees of freedom may perform their computations in ways that are completely inscrutable to the theorist. Some may take delight in such a result, but there is reason to be concerned by it. It can be argued that getting a machine to perform intelligently is more important than understanding how it does so. If a magic procedure—say for learning—did in fact lead to the level of performance desired, despite our inability to understand the resulting computation, that would of course be a landmark accomplishment. But to expect this kind of breakthrough is just the sort of naiveté referred to in the first paragraph of the chapter. We now have enough disappointing experience to expect that any particular insight is going to take us a *very small fraction* of the way to the kind of truly intelligent mechanisms we seek. The only way to reasonably expect to make progress is by chaining together many such small steps. And the only way to chain together these steps is to *understand* at the end of each one where we are, how we got there, and why we got no further, so we can make an informed guess as to how to take the next small step. A "magic" step is apt to be a *last* step; it is fine, as long as it takes you exactly where you want to go.

HARMONY AND PROBABILITY

The Harmony Function

The preceding section described how environmental states and knowledge are represented in harmony theory. The use of this knowledge in completing representations of environmental states is governed by the harmony function, which, as discussed in Section 1, measures the self-consistency of any state of a harmony model. I will now discuss the properties required of a harmony function and present the particular function I have studied.

A state of the cognitive system is determined by the values of the lower and upper level nodes. Such a state is determined by a pair $(\mathbf{r}, \mathbf{a})$ consisting of a representation vector $\mathbf{r}$ and an activation vector $\mathbf{a}$. A harmony function assigns a real number $H_K(\mathbf{r}, \mathbf{a})$ to each such state.

The harmony function has as parameters the set of knowledge vectors and their strengths: $\{(\mathbf{k}_\alpha,\sigma_\alpha)\}$; I will call this the *knowledge base* $\mathbf{K}$.

The basic requirement on the harmony function H is that it be *additive under decompositions* of the system.[13] This means that if a network can be partitioned into two unconnected networks, as in Figure 12, the harmony of the whole network is the sum of the harmonies of the parts:

$$H(\mathbf{r},\mathbf{a}) = H(\mathbf{r}_1,\mathbf{a}_1) + H(\mathbf{r}_2,\mathbf{a}_2).$$

In this case, the knowledge and representational feature nodes can each be broken into two subsets so that the knowledge atoms in subset 1 all have 0 connections with the representational features in subset 2, and vice versa. Corresponding to this partition of nodes there is a decomposition of the vectors $\mathbf{r}$ and $\mathbf{a}$ into the pieces $\mathbf{r}_1,\mathbf{r}_2$ and $\mathbf{a}_1,\mathbf{a}_2$.

The harmony function I have studied (recall Figure 7) is

$$H_{\mathbf{K}}(\mathbf{r},\mathbf{a}) = \sum_\alpha \sigma_\alpha\, a_\alpha\, h_\kappa(\mathbf{r},\mathbf{k}_\alpha). \tag{1}$$

Here, $h_\kappa(\mathbf{r},\mathbf{k}_\alpha)$ is the harmony contributed by activating atom α, given the current representation $\mathbf{r}$. I have taken this to be

$$h_\kappa(\mathbf{r},\mathbf{k}_\alpha) = \frac{\mathbf{r}\cdot\mathbf{k}_\alpha}{|\mathbf{k}_\alpha|} - \kappa.$$

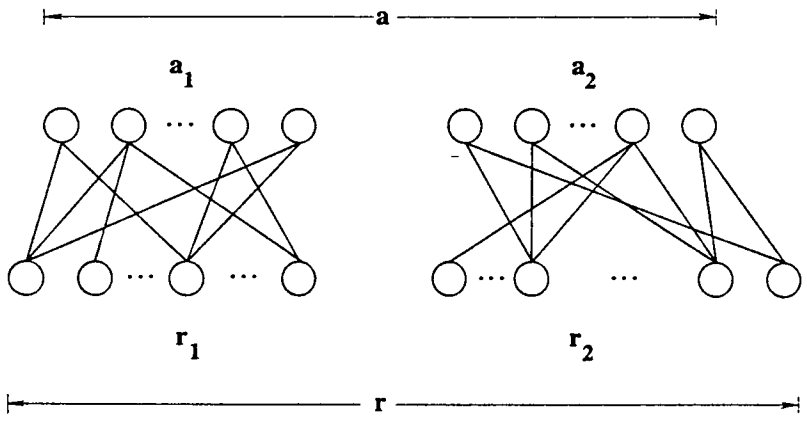

FIGURE 12. A decomposable harmony network.

[13] In physics, one says that H must be an *extensive quantity*.

The vector inner product (see Chapter 9) is defined by

$$\mathbf{r}\cdot\mathbf{k}_\alpha = \sum_i r_i (\mathbf{k}_\alpha)_i$$

and the norm [14] is defined by

$$|\mathbf{k}_\alpha| = \sum_i |(\mathbf{k}_\alpha)_i|.$$

I will now comment on these definitions.

First note that this harmony function $H_{\mathbf{K}}$ is a sum of terms, one for each knowledge atom, with the term for atom α depending only on those representation variables r_i to which it has nonzero connection $(\mathbf{k}_\alpha)_i$. Thus $H_{\mathbf{K}}$ satisfies the additivity requirement.

The contribution to H of an inactive atom is zero. The contribution of an active atom α is the product of its strength and the consistency between its knowledge vector $\mathbf{k}_\alpha$ and the representation vector $\mathbf{r}$; this is measured by the function $h_\kappa(\mathbf{r}, \mathbf{k}_\alpha)$. The parameter κ always lies in the interval $(-1,1)$. When $\kappa = 0$, $h_\kappa(\mathbf{r}, \mathbf{k}_\alpha)$ is the number of representational features whose values agree with the corresponding value in the knowledge vector minus the number that disagree. This gives the simplest harmony function, the one described in Section 1. The trouble is that according to this measure, if over 50% of the knowledge vector $\mathbf{k}_\alpha$ agrees with $\mathbf{r}$, the harmony is raised by activating atom α. This is a pretty weak criterion of matching, and sometimes it is important to be able to have a more stringent criterion than 50%. As κ goes from -1 through 0 towards 1, the criterion goes from 0% through 50% towards 100%. In fact it is easy to see that the criterial fraction is $(1 + \kappa)/2$. The total harmony will be raised by activating any atom for which the number of representational features on which the atom's knowledge vector agrees with the representation vector exceeds this fraction of the total number of possible agreements ($|\mathbf{k}_\alpha|$).

An important limit of the theory is $\kappa \to 1$. In this limit, the criterion approaches perfect matching. For any given harmony model, perfect matching is required by any κ greater than some definite value less than 1 because there is a limit to how close to 100% matching one can achieve with a finite number of possible matches. Indeed it is easy to compute that if n is the largest number of nonzero connections to any atom in a model (the maximum of $|\mathbf{k}_\alpha|$), then the only way to exceed a

---

[14] This is the so-called L_1 norm, which is different from the L_2 norm defined in Chapter 9. For each p in $(0,\infty)$ the L_p norm of a vector $\mathbf{v}$ is defined by

$$|\mathbf{v}|_p = \left[\sum_i |v_i|^p\right]^{1/p}.$$

criterion of $1 - 2/n$ is with a perfect match. Any κ value greater than this will place the model in what I will call the *perfect matching limit*. Note that since harmony theory is probabilistic, even in the perfect matching limit, atoms will sometimes become active even when they do not match the current representation perfectly; the closer the match, the more likely they will be active.

By choosing $+1$ and -1 as the binary values for representational features, we have ensured that the product $(\mathbf{k}_\alpha)_i r_i$ will be $+1$ if the knowledge vector agrees with r_i, -1 if it disagrees, and 0 if it doesn't specify a value for feature i. The maximum value that can be attained by $\mathbf{k}_\alpha \cdot \mathbf{r}$ is $|\mathbf{k}_\alpha|$, the number of nonzero connections to node α, irrespective of whether those connections are $+$ or $-$.

In fact, this harmony function is *invariant under the exchange of* $+$ *and* $-$ *at any representation node.* That is, simultaneously flipping the signs of r_i and $(\mathbf{k}_\alpha)_i$ for all α leaves the value of $H_{\mathbf{K}}(\mathbf{r}, \mathbf{a})$ unchanged, for every $\mathbf{a}$. This symmetry was deliberately inserted into the general harmony function because I could think of no principled reason to break it. If a systematic bias in the representation variables toward one of the binary values is to be built in from the outset, how large should the bias be? It seemed reasonable to start the theory in a symmetric way, unbiased toward either value. Of course a bias *can* be inserted through the *knowledge* $\mathbf{K}$. To take an extreme example, if the value of feature i is $+$ in all knowledge atoms, i.e., $(\mathbf{k}_\alpha)_i = +$ for all α, then the ith feature r_i will be strongly biased toward $+$.

There is nothing sacred about the values $+1$ and -1 in this theory. The values 1 and 0, for example, could be used as well. The preceding harmony function can easily be rewritten to give the same harmony values when $\mathbf{r}$ is changed from the $\{+1,-1\}$ form to the $\{1,0\}$ form. The underlying invariance under sign change would however be transformed into a more complicated invariance.

Estimating Probabilities With the Harmony Function

In Section 1, I suggested that a cognitive system performing the completion task could use a harmony function for estimating the probabilities of values for unknown variables. In fact, Point 9 asserted that the estimated probability of a set of values for unknown variables was an exponential function of the corresponding harmony value:

$$\text{probability} \propto e^{H/T}. \tag{2}$$

It is this relationship that establishes the mapping with statistical physics. In this section and the next, the relationship between harmony

and probability is analyzed. In this section I will point out that *if* probabilities are to be estimated using H, then the exponential relationship of Equation 2 should be used. In the next section I adapt an argument of Stuart Geman (personal communication, 1984) to show that, starting from the extremely general probabilistic assumption known as the *principle of maximum missing information*, both Equation 2 and the form of the harmony function (Equation 1) can be jointly derived.

What we know about harmony functions in general is that they are additive under network decomposition. If a harmony network consists of two unconnected components, the harmony of any given state of the whole network is the *sum* of the harmonies of the states of the component networks. In the case of such a network, what is required of the *probability* assigned to the state? I claim it should be the *product* of the probabilities assigned to the states of the component networks. The meaning of the unconnectedness is that the knowledge used in the inference process does not relate the features in the two networks to each other. Thus the results of inference about these two sets of features should be *independent*. Since the probabilities assigned to the states in the two networks should be independent, the probability of their joint occurrence—the state of the network as a whole—should be the product of their individual probabilities.

In other words, *adding* the *harmonies* of the components' states should correspond to *multiplying* the *probabilities* of the components' states. The exponential function of Equation 2 establishes just this correspondence. It is a mathematical fact that the *only* continuous functions f that map addition into multiplication,

$$f(x + y) = f(x)\, f(y)$$

are the exponential functions,

$$f(x) = a^x$$

for some positive number a. Equivalently, these functions can be written

$$f(x) = e^{x/T}$$

for some value T (where $T = 1/\ln a$).

This general argument leaves undetermined the value of T, the computational temperature. However several observations about the value of T can be made.

First, the *sign* of T must be positive, for otherwise *greater* harmony would correspond to *smaller* probability.

For the second observation, consider a cognitive system a that estimates its environmental probability distribution with a certain value for

T_a and a certain harmony function H_a. Then given any other positive temperature T_b, we could hypothesize another cognitive system b using that computational temperature and the modified harmony function $H_b = (T_b/T_a)\, H_a$. Both cognitive systems would have the same estimates of environmental probabilities since $H_b/T_b = H_a/T_a$. Thus their behavior on the completion task would be indistinguishable.

Thus, the *magnitude* of T is only meaningful *once a specific scale has been set for* H. This means that if H is being *learned* by the system, rather than programmed in by the modeler, then any convenient choice of T will do; the choice simply determines the scale of H that the system will learn.

The third observation refines the second. A convenient way of expressing Equation 2 is to use the *likelihood ratio* of two states $\mathbf{s}_1$ and $\mathbf{s}_2$:

$$\frac{\text{prob}(\mathbf{s}_1)}{\text{prob}(\mathbf{s}_2)} = e^{[H(\mathbf{s}_1)-H(\mathbf{s}_2)]/T}. \tag{3}$$

Thus, T *sets the scale for those differences in harmony that correspond to significant differences in probability.* (It is understood here that "differences" in harmony are measured by *subtraction* while "differences" in probability are measured by *division*.) The smaller the value of T, the smaller the harmony differences that will correspond to significant likelihood ratios. Thus, once a scale of H has been fixed, decreasing the value of T makes the probability distribution *more sharply peaked.* In fact, Equation 3 can be rewritten

$$\frac{\text{prob}(\mathbf{s}_1)}{\text{prob}(\mathbf{s}_2)} = \left[e^{H(\mathbf{s}_1)-H(\mathbf{s}_2)}\right]^{1/T}.$$

If state $\mathbf{s}_1$ has greater harmony than $\mathbf{s}_2$, the likelihood ratio at $T = 1$ will be the number in square brackets, a number greater than one; as T goes to zero this number gets raised to higher and higher powers so that the likelihood ratio goes to infinity. In other words, compared to T, the fixed difference in harmony between the two states looks larger and larger as T gets smaller and smaller.

In the preceding argument, the exponential functions emerged as the only continuous functions mapping addition into multiplication. Of course we could consider discontinuous functions, one example being the limit as $T \to 0$ of the exponential. In this limit, the estimated probability of all states is zero, except the ones with maximal harmony. If there are several states with exactly the same maximal harmony, in the zero temperature limit they will all end up with equal, nonzero probability. This probability distribution will be called *the zero temperature distribution.* It does not correspond to an exponential distribution,

but it can be obtained as the limit of exponential distributions; in fact, the zero-temperature limit plays a major role in the theory since the states of maximal harmony are the best answers to completion problems.

THE COMPETENCE, REALIZABILITY, AND LEARNABILITY THEOREMS

In this section, the mathematical results that currently form the core of harmony theory are informally described. A formal presentation may be found in the Appendix.

The Competence Theorem

In harmony theory, a cognitive system's knowledge is encoded in its knowledge atoms. Each atom represents a pattern of values for a few features describing environmental states, values that sometimes co-occur in the system's environment. The strengths of the atoms encode the frequencies with which the different patterns occur in the environment. The atoms are used to estimate the probabilities of events in the environment.

Suppose then that a particular cognitive system is capable of observing the frequency with which each pattern in some pre-existing set $\{\mathbf{k}_\alpha\}$ occurs in its environment. (The larger the set $\{\mathbf{k}_\alpha\}$, the greater is the potential power of this cognitive system.) Given the frequencies of these patterns, how should the system estimate the probabilities of environmental events? What probability distribution should the system guess for the environment?

There will generally be many possible environmental distributions that are consistent with the known pattern frequencies. How can one be selected from all these possibilities?

Consider a simple example. Suppose there are only two environmental features in the representation, r_1 and r_2, and that the system's only information is that the pattern $r_1 = +$ occurs with a frequency of 80%. There are infinitely many probability distributions for the four environmental events $(r_1,r_2) \in \{(+,+)\ (+,-)\ (-,+)\ (-,-)\}$ that are consistent with the given information. For example, we know nothing about the relative likelihood of the two events $(+,+)$ and $(+,-)$; all we know is that together their probability is .80 .

One respect in which the possible probability distributions differ is in their degree of homogeneity. A distribution P in which $P(+,+) = .7$

and $P(+,-) = .1$ is less homogeneous than one for which both these events have probability .4.

Another way of saying this is that the *uncertainty* associated with the second distribution is greater than that of the first. In Shannon's (1948/1963) terms, if the second, more homogeneous, distribution applies, then at any given moment there is a greater amount of *missing information* about the current state of the environment than there is if the more inhomogenous distribution applies. Shannon's formula for the missing information of a probability distribution P is

$$-\sum_x P(x)\ \ln P(x).$$

Thus the missing information in the inhomogeneous probabilities $\{.7, .1\}$ is

$$-\Big[.7\ln(.7) + .1\ln(.1)\Big] = .48$$

while the missing information in the homogeneous probabilities $\{.4, .4\}$ is

$$-\Big[.4\ln(.4) + .4\ln(.4)\Big] = .73.$$

The cognitive system's information on the frequency of patterns contains some information about any lack of homogeneity in the environmental distribution. One principle for guessing the environmental distribution is to select, of all probability distributions that are consistent with the known frequencies, the one that is most homogenous; the one that supposes the environment to have no more inhomogeneity than is needed to account for the known information. This principle can be formalized as the *principle of maximal missing information*; it is often used to extrapolate from some given statistical information to an estimate for an entire probability distribution (Christensen, 1981; Levine & Tribus, 1979).

For the simple example discussed above, the principle of maximal missing information implies that the cognitive system should estimate the environmental distribution to be $P(+,+) = P(+,-) = .40$, $P(-,+) = P(-,-) = .10$. This distribution is inhomogeneous with respect to the first feature, r_1, because it *must* be to account for the known fact that $P(r_1 = +) = .80$. It is homogeneous in the second feature, r_2, because it *can* be without violating any known information. The justification for choosing this distribution is that there is not enough given information to justify selecting any other distribution with less missing information.

In the general case, one can use the formula for missing information to derive the distribution with maximal missing information that is

consistent with the observed frequencies of the patterns $\mathbf{k}_\alpha$. The result is a probability distribution I will call π:

$$\pi(\mathbf{r}) \propto e^{U(\mathbf{r})}$$

where the function U is defined by

$$U(\mathbf{r}) = \sum_\alpha \lambda_\alpha \chi_\alpha(\mathbf{r}).$$

The values of the real parameters λ_α (one for each atom) are constrained by the known pattern frequencies; they will shortly be seen to be proportional to the atom strengths, σ_α, the system should use for modeling the environment. The value of $\chi_\alpha(\mathbf{r})$ is simply 1 when the environmental state $\mathbf{r}$ includes the pattern $\mathbf{k}_\alpha$ defining atom α, and 0 otherwise.

Now that we have a formula for estimating the probability of an environmental state, we can in principle perform the completion task. An input for this task is a set of values for some of the features. The best completion is formed by assigning values to the unknown features so that the resulting vector $\mathbf{r}$ represents the most probable environment state, as estimated by π.

It turns out that the completions performed in this way are *exactly* the same as those that would be formed by using the same procedure with the different distribution

$$p(\mathbf{r},\mathbf{a}) \propto e^{H(\mathbf{r},\mathbf{a})}.$$

Here, H is the harmony function defined previously, where the strengths are

$$\sigma_\alpha = \frac{\lambda_\alpha}{1-\kappa}$$

and κ is any value satisfying

$$1 > \kappa > 1 - 2/\left[\max_\alpha |\mathbf{k}_\alpha|\right].$$

(This condition on κ is the *exact matching limit* defined earlier.)

In passing from $\pi(\mathbf{r})$ to $p(\mathbf{r},\mathbf{a})$, new variables have been introduced: the activations $\mathbf{a}$. These serve to eliminate the functions χ_α from the formula for estimating probabilities, which will be important shortly when we try to design a device to actually perform the completion computation. The result is that in addition to filling in the unknown features in $\mathbf{r}$, all the activations in $\mathbf{a}$ must be filled in as well. In other words, to perform the completion, the cognitive system must find those

values of the unknown r_i and those values of the a_α that together maximize the harmony $H(\mathbf{r},\mathbf{a})$ and thereby maximize the estimated probability $p(\mathbf{r},\mathbf{a})$.

This discussion is summarized in the following theorem:

Theorem 1: Competence. Suppose a cognitive system can observe the frequency of the patterns $\{\mathbf{k}_\alpha\}$ in its environment. The probability distribution with the most Shannon missing information that is consistent with the observations is

$$\pi(\mathbf{r}) \propto e^{U(x)}$$

with U defined as above. The maximum-likelihood completions of this distribution are the same as those of

$$p(\mathbf{r},\mathbf{a}) \propto e^{H(\mathbf{r},\mathbf{a})}$$

with the harmony function defined above.

This theorem describes how a cognitive system *should* perform completions, according to some mathematical principles for statistical extrapolation and inference. In this sense, it is a *competence* theorem. The obvious next question is: Can we design a system that will really compute completions according to the specifications of the competence theorem?

The "Physics Analogy"

It turns out that designing a machine to do the required computations is a relatively straightforward application of a computational technique from statistical physics. It is therefore an appropriate time to discuss the "analogy" to physics that is exploited in harmony theory.

Why is the relation between probability and harmony expressed in the competence theorem the same as the relation between probability and energy in statistical physics? The mapping between statistical physics and inference that is being exploited is one that has been known for a long time.

The second law of thermodynamics states that as physical systems evolve in time, they will approach conditions that maximize randomness or *entropy*, subject to the constraint that a few conserved quantities like the systems' energy must always remain unchanged. One of the triumphs of statistical mechanics was the understanding that this law is the macroscopic manifestation of the underlying microscopic description

of matter in terms of constituent particles. The particles will occupy various states and the macroscopic properties of a system will depend on the probabilities with which the states are occupied. The randomness or entropy of the system, in particular, is the homogeneity of this probability distribution. It is measured by the formula

$$-\sum_x P(x) \ln P(x).$$

A system evolves to maximize this entropy, and, in particular, a system that has come to equilibrium in contact with a large reservoir of heat will have a probability distribution that maximizes entropy subject to the constraint that its energy have a fixed average value.

Shannon realized that the homogeneity of a probability distribution, as measured by the microscopic formula for entropy, was a measure of the missing information of the distribution. He started the book of information theory with a page from statistical mechanics.

The competence theorem shows that the exponential relation between harmony and probability stems from maximizing missing information subject to the constraint that given information be accounted for. The exponential relation between energy and probability stems from maximizing entropy subject to a constraint on average energy. The physics analogy therefore stems from the fact that entropy and missing information share exactly the same relation to probability. It is not surprising that the theory of information processing should share formal features with the theory of statistical physics.

Shannon began a mapping between statistical physics and the theory of information by mapping entropy onto information content. Harmony theory extends this mapping by mapping self-consistency (i.e., harmony) onto energy. In the next subsection, the mapping will be further extended to map stochasticity of inference (i.e., computational temperature) onto physical temperature.

The Realizability Theorem

The mapping with statistical physics allows harmony theory to exploit a computational technique for studying thermal systems that was developed by N. Metropolis, M. Rosenbluth, A. Rosenbluth, A. Teller, and E. Teller in 1953. This technique uses stochastic or "Monte Carlo" computation to simulate the probabilistic dynamical system under study. (See Binder, 1979.)

The procedure for simulating a physical system at temperature T is as follows: The variables of the system are assigned random initial

values. One by one, they are updated according to a stochastic rule: The probability of assigning a new value x to the variable is proportional to $e^{H_x/T}$, where H_x is (minus) the energy the system would have if the value x were chosen. Thus the higher T, the more random are the decisions. As the computation proceeds, the probability that the system is in state s at any moment becomes proportional to the desired value, $e^{H(s)/T}$.

Adapting this technique to the computations of harmony theory leads, through an analysis described in the Appendix, to the following theorem. It defines the machine *harmonium* that realizes the theory of completions expressed in Theorem 1.

Theorem 2: Realizability. In the graphical representation of a harmony system (see Figure 13) let each node denote a processor. Each feature node processor can have a value of $+1$ or -1, and each knowledge atom a value of 1 or 0 (its activation). Let the input to a completion problem be specified by assigning the given feature nodes their correct values; these are fixed throughout the computation. All other nodes repeatedly update their values during the computation. The features not specified in the input are assigned random initial values, and the knowledge atoms initially all have value 0. Let each node stochastically update its value according to the rule:

$$\text{prob}(\text{value} = 1) = \frac{1}{1 + e^{-I/T}}$$

where T is a global system parameter and I is the "input" to the node from the other nodes attached to it (defined below). All the

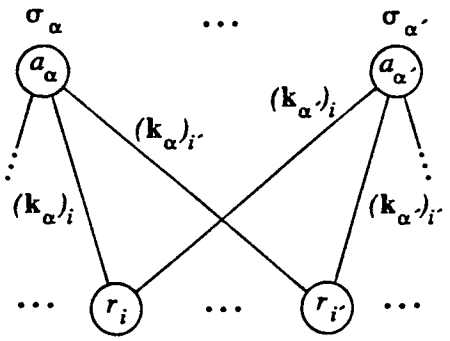

FIGURE 13. A general harmony graph.

nodes in the upper layer update in parallel, then all the nodes in the lower layer update in parallel, and so on alternately throughout the computation. During the update process, T starts out at some positive value and is gradually lowered. If T is lowered to 0 sufficiently slowly, then asymptotically, with probability 1, the system state forms the best completion (or one of the best completions if there are more than one that maximize harmony).

To define the input I to each node, it is convenient to assign to the link in the graph between atom α and feature i a weight $W_{i\alpha}$ whose sign is that of the link and whose magnitude is the strength of the atom divided by the number of links to the atom:

$$W_{i\alpha} = (\mathbf{k}_\alpha)_i \frac{\sigma_\alpha}{|\mathbf{k}_\alpha|}.$$

Using these weights, the input to a node is essentially the weighted sum of the values of the nodes connected to it. The exact definitions are

$$I_i = 2 \sum_\alpha W_{i\alpha} a_\alpha$$

for feature nodes, and

$$I_\alpha = \sum_i W_{i\alpha} r_i - \kappa$$

for knowledge atoms.

The formulae for I_i and I_α are both derived from the fact that the input to a node is precisely the harmony the system would have if the given node were to choose the value 1 minus the harmony resulting from not choosing 1. The factor of 2 in the input to a feature node is in fact the difference $(+1) - (-1)$ between its possible values. The term κ in the input to an atom comes from the κ in the harmony function; it is a threshold that must be exceeded if activating the atom is to increase the harmony.

The stochastic decision rule can be understood with the aid of Figure 14. If the input to the node is large and positive (i.e., selecting value 1 would produce much greater system harmony), then it will almost certainly choose the value 1. If the input to the node is large and negative (i.e., selecting value 1 would produce much lower system harmony), then it will almost certainly *not* choose the value 1. If the input to the node is near zero, it will choose the value 1 with a probability near .5. The width of the zone of random decisions around zero input is larger the greater is T.

The process of gradually lowering T can be thought of as *cooling the*

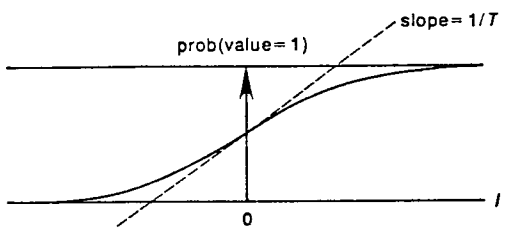

FIGURE 14. The relation between the input I to a harmonium processor node and the probability the processor will choose the value 1.

randomness out of the initial system state. In the limit that $T \rightarrow 0$, the zone of random decisions shrinks to zero and the stochastic decision rule becomes the deterministic linear threshold rule of perceptrons (Minsky & Papert, 1969; see Chapter 2). In this limit, a node will always select the value with higher harmony. At nonzero T, there is a finite probability that the node will select the value with lower harmony.

Early in a given computation, the behavior of the processors will be highly random. As T is lowered, gradually the decisions made by the processors will become more systematic. In this way, parts of the network gradually assume values that become stable; the system commits itself to decisions as it cools; it passes from fluid behavior to the rigid adoption of an answer. The decision-making process resembles the crystallization of a liquid into a solid.

Concepts from statistical physics can in fact usefully be brought to bear on the analysis of decision making in harmony theory, as we shall see in the next section. As sufficient understanding of the computational effects of different cooling procedures emerges, the hope is that harmony theory will acquire an account of how a cognitive system can regulate its own computational temperature.

Theorem 2 describes how to find the best completions by lowering to zero the computational temperature of a parallel computer—harmonium—based on the function H. Harmonium thus realizes the second half of the competence theorem, which deals with optimal completions. But Theorem 1 also states that estimates of environmental probabilities are obtained by exponentiating the function U. It is also possible to build a stochastic machine based on U that is useful for simulating the environment. I will call this the *simulation machine.*

Figure 15 shows the portion of a harmonium network involving the atom α, and the corresponding portion of the processor network for the corresponding simulation machine. The knowledge atom with strength

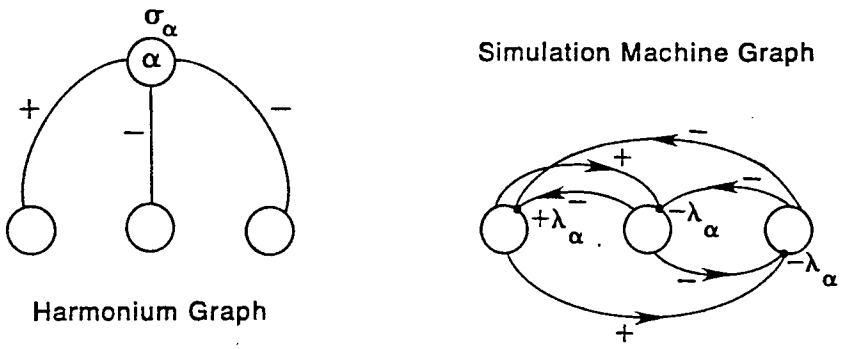

FIGURE 15. The graph for a one-atom harmony function and the graph for the corresponding U function. In the latter, there are only feature nodes. Each feature node has a single input point labeled $\pm\lambda$, where the sign is the same as that assigned to the feature by the knowledge atom. Into this input point come links from all the other features assigned values by the knowledge atom. The label on each arc leaving a feature is the same as the value assigned to that feature by the knowledge atom.

σ_α and feature pattern $(+,-,-)$ is replaced by a set of connections between pairs of features. In accordance with Theorem 1, $\lambda_\alpha = \sigma_\alpha(1-\kappa)$. For every atom α connected to a given feature in harmonium, in the simulation machine there is a corresponding input point on that feature, labeled with λ_α.

The update rule for the simulation machine is the same as for harmonium. However, only one node can update at a time, and the definition of the input I to a node is different.[15] The input to a feature node is the sum of the inputs coming through all input points to the node. If an input point on node i is labeled $\pm\lambda_\alpha$, then the input coming to i through that point is $\pm\lambda_\alpha$ if the values of all the nodes connected to i agree with the label on the arc connecting it to i, and zero otherwise.

If the simulation machine is operated at a fixed temperature of 1, the probability that it will be found in state $\mathbf{r}$ asymptotically becomes proportional to $e^{U(\mathbf{r})/1}$. By Theorem 1, this is the cognitive system's estimate $\pi(\mathbf{r})$ of the probability that the environment will be in the state represented by $\mathbf{r}$. Thus running this machine at temperature 1 gives a simulation of the environment. As we are about to see, this will turn out to be important for learning.

The general type of search procedure used by harmonium, with a random "thermal noise" component that is reduced during the computation, has been used to find maxima of functions other than harmony

[15] Analogously to harmonium, the input to a node is the value U would have if the node adopted the value $+1$, minus the value U it would have if it adopted the value -1.

functions. Physicists at IBM independently applied the technique, under the name *simulated annealing*, to both practical computer design problems and classical maximization problems (Kirkpatrick, Gelatt, & Vecchi, 1983). Benchmarks of simulated annealing against other search procedures have produced mixed results (Aragon, Johnson, & McGeoch, 1985).

The contribution of harmony theory is not so much the search procedure for finding maxima of H, but rather the function H itself. Theorem 2 is important: It describes a statistical dynamical system that performs completions; it gives an implementation-level description of a kind of completion machine. But Theorem 1 is more central: It gives a high, functional-level characterization of the performance of the system—says what the machine does—and introduces the concept of harmony. More central to the theory also is Theorem 3, which says how the harmony function can be tuned with experience.

The Learnability Theorem

Performing the completion task in different environments calls for different knowledge. In the formalism of Theorem 1, a given cognitive system is assumed to be capable of observing the frequency in its environment of a predetermined set of feature patterns. What varies for a given cognitive system across environments is the *frequencies* of the patterns; this manifests itself in the variation across environments of the *strengths* of the knowledge atoms representing those patterns.

> *Theorem 3: Learnability.* Suppose states of the environment are selected according to the probability distribution defining that environment, and each state is presented to a cognitive system. Then there is a procedure for gradually modifying the strengths of the knowledge atoms that will converge to the values required by Theorem 1.

The basic idea of the learning procedure is simple. Whenever one of the patterns the cognitive system can observe is present in a stimulus from the environment, the parameter associated with that pattern is incremented. In harmonium, this means that whenever a knowledge atom matches a stimulus, its strength increases by a small amount $\Delta\sigma$. In the simulation machine, this means that the λ parameter on all the connections corresponding to that atom must be incremented by $\Delta\lambda = \Delta\sigma(1 - \kappa)$. In this sense, an atom corresponds to a *memory*

trace of a feature pattern, and the strength of the atom is the strength of the trace: greater the more often it has been experienced.

There is an error-correcting mechanism in the learning procedure that decrements parameters when they become too large. Intermixed with its *observation* of the environment, the cognitive system must perform *simulation* of the environment. As discussed above, this can be done by running the simulation machine at temperature 1 without input from the environment. During simulation, patterns that appear in the feature nodes produce exactly the *opposite* effect as during environmental observation, i.e., a *decrement* in the corresponding parameters.

Harmonium can be used to approximate the simulation machine. By running harmonium at temperature 1, without input, states are visited with a probability of e^H, which approximates the probabilities of the simulation machine, e^U.[16] When harmonium is used to approximately simulate the environment, every time an atom matches the feature vector its strength is *decremented* by $\Delta\sigma$.

This error-correcting mechanism has the following effect. The strength of each atom will stabilize when it gets (on the average) incremented during environmental observation as often as it gets decremented during environmental simulation. If environmental observation and simulation are intermixed in equal proportion, the strength of each atom will stabilize when its pattern appears as often in simulation as in real observation. This means the simulation is as veritical as it can be, and that is why the procedure leads to the strengths required by the competence theorem.

DECISION-MAKING AND FREEZING

The Computational Significance of Phase Transitions

Performing the completion task requires simultaneously satisfying many constraints. In such problems, it is often the case that it is easy to find "local" solutions that satisfy some of the constraints but very difficult to find a global solution that simultaneously satisfies the maximum number of constraints. In harmony theory terms, often there are *many* completions of the input that are *local* maxima of H, in which some knowledge atoms are activated, but very *few* completions that are *global* maxima, in which many atoms can be simultaneously activated.

When harmonium solves such problems, initially, at high

[16] Theorem 1 makes this approximation precise: These two distributions are not equal, but the maximum-probability states are the same for any possible input.

temperatures, it occupies states that are local solutions, but finally, at low temperatures, it occupies only states that are global solutions. If the problem is well posed, there is only one such state.

Thus the process of solving the problem corresponds to the passage of the harmonium dynamical system from a high-temperature phase to a low-temperature phase. An important question is: *Is there a sharp transition between these phases?* This is a "freezing point" for the system, where major decisions are made that can only be undone at lower temperatures by waiting a very long time. It is important to cool slowly through phase transitions, to maximize the chance for these decisions to be made properly; then the system will relatively quickly find the global harmony maximum without getting stuck for very long times in local maxima.

In this section, I will discuss an analysis that suggests that phase transitions *do* exist in very simple harmony theory models of decision-making. In the next section, a more complex model that answers simple physics questions will furnish another example of a harmony system that seems to possess a phase transition.[17]

The cooling process is an essentially new feature of the account of cognitive processing offered by harmony theory. To analyze the implications of cooling for cognition, it is necessary to analyze the temperature dependence of harmony models. Since the mathematical framework of harmony theory significantly overlaps that of statistical mechanics, general concepts and techniques of thermal physics can be used for this analysis. However, since the *structure* of harmony models is quite different from the structure of models of real physical systems, specific results from physics cannot be carried over. New ideas particular to cognition enter the analysis; some of these will be discussed in a later section on the macrolevel in harmony theory.

Symmetry Breaking

At high temperatures, physical systems typically have a *disordered* phase, like a fluid, which dramatically shifts to a highly *ordered* phase,

[17] It is tempting to identify freezing or "crystallization" of harmonium with the phenomenal experience of sudden "crystallization" of scattered thoughts into a coherent form. There may even be some usefulness in this identification. However, it should be pointed out that since cooling should be slow at the freezing point, in terms of iterations of harmonium, the transition from the disordered to the ordered phase may *not* be sudden. If iterations of harmonium are interpreted as real cognitive processing time, this calls into question the argument that "sudden" changes as a function of *temperature* correspond to "sudden" changes as a function of real time.

like a crystal, at a certain freezing temperature. In the low-temperature phase, a single ordered configuration is adopted by the system, while at high temperatures, parts of the system shift independently among pieces of ordered configurations so that the system as a whole is a constantly changing, disordered blend of pieces of different ordered states.

Thus we might expect that at high temperatures, the states of harmonium models will be shifting blends of pieces of reasonable completions of the current input; it will form *locally* coherent solutions. At low temperatures (in equilibrium), the model will form completions that are *globally* coherent.

Finding the best solution to a completion problem may involve fine discriminations among states that all have high harmonies. There may even be several completions that have exactly the same harmonies, as in interpreting ambiguous input. This is a useful case to consider, for in an ordered phase, harmonium must at any time construct one of these "best answers" in its pure form, without admixing parts of other best answers (assuming that such mixtures are not themselves best answers, which is typically the case). In physical terminology, *the system must break the symmetry* between the equally good answers in order to enter the ordered phase. One technique for finding phase transitions is to look for critical temperatures above which symmetry is respected, and below which it is broken.

An Idealized Decision

This suggests we consider the following idealized decision-making task. Suppose the environment is always in one of two states, A and B, with equal probability. Consider a cognitive system performing the completion task. Now for some of the system's representational features, these two states will correspond to the same feature value. These features do not enter into the decision about which state the environment is in, so let us remove them. Now the two states correspond to opposite values on all features. We can assume without loss of generality that for each feature, + is the value for A, and − the value for B (for if this were not so we could redefine the features, exploiting the symmetry of the theory under flipping signs of features). After training in this environment, the knowledge atoms of our system each have either all + connections or all − connections to the features.

To look for a phase transition, we see if the system can break symmetry. We give the system a completely ambiguous input: no input at all. It will complete this to either the all-+ state, representing A, or the all-− state, representing B, each outcome being equally likely.

Observing the harmonium model we see that for high temperatures, the states are typically blends of the all-+ and all-− states. These blends are not themselves good completions since the environment has no such states. But at low temperatures, the model is almost always in one pure state or the other, with only short-lived intrusions on a feature or two of the other state. It is equally likely to cool into either state and, given enough time, will flip from one state to the other through a sequence of (very improbable) intrusions of the second state into the first. The transition between the high- and low-temperature phases occurs over a quite narrow temperature range. At this freezing temperature, the system drifts easily back and forth between the two pure states.

The harmonium simulation gives empirical evidence that there is a critical temperature below which the symmetry between the interpretations of ambiguous input is broken. There is also analytic evidence for a phase transition in this case. This analysis rests on an important concept from statistical mechanics: the thermodynamic limit.

The Thermodynamic Limit

Statistical mechanics relates microscopic descriptions that view matter as dynamical systems of constituent particles to the macrolevel descriptions of matter used in thermodynamics. Thermodynamics provides a good approximate description of the bulk properties of systems containing an extremely large number of particles. The *thermodynamic limit* is a theoretical limit in which the number of particles in a statistical mechanical system is taken to infinity, keeping finite certain aggregate properties like the system's density and pressure. It is in this limit that the microtheory provably admits the macrotheory as a valid approximate description.

The thermodynamic limit will later be seen to relate importantly to the limit of harmony theory in which symbolic macro-accounts become valid. But for present purposes, it is relevant to the analysis of phase transitions. One of the important insights of statistical mechanics is that *qualitative* changes in thermal systems, like those characteristic of genuine phase transitions, cannot occur in systems with a finite number of degrees of freedom (e.g., particles). It is only in the thermodynamic limit that phase transitions can occur.

This means that an analysis of freezing in the idealized-decision model must consider the limit in which the number of features and knowledge atoms go to infinity. In this limit, certain approximations become valid that suggest that indeed there *is* a phase transition.

Robustness of Coherent Interpretation

To conclude this section, let me point out the significance of this simple decision-making system. Harmony theory started out to design an engine capable of constructing coherent interpretations of input and ended up with a class of thermal models realized by harmonium. We have just seen that the resulting models are capable of taking a completely ambiguous input and nonetheless constructing a completely coherent interpretation (by cooling below the critical temperature). This suggests a robustness in the drive to construct coherent interpretations that should prove adequate to cope with more typical cases characterized by less ambiguity but greater complexity. The greater complexity will surely hamper *our* attempts to analyze the models' performance; it remains to be seen whether greater complexity will hamper the models' ability to construct coherent interpretations. With this in mind, we now jump to a much more complex decision-making problem: the qualitative analysis of a simple electric circuit.

AN APPLICATION: ELECTRICITY PROBLEM SOLVING

Theoretical context of the model. In this section I show how the framework of harmony theory can be used to model the *intuition* that allows experts to answer, without any conscious application of "rules," questions like that posed in Figure 16. Theoretical conceptions of how such problems are answered plays an increasingly significant role in the design of instruction. (For example, see the new journal, *Cognition and*

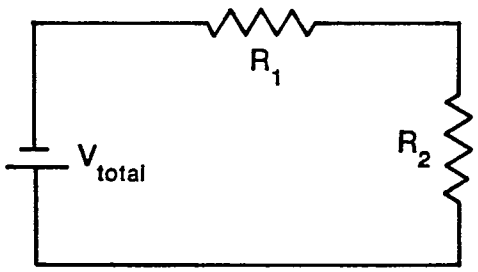

FIGURE 16. If the resistance of R_2 is increased (assuming that V_{total} and R_1 remain the same), what happens to the current and voltage drops?

Instruction, and Ginsburg, 1983.) Even such simple problems as that of Figure 16 have important instructional implications (Riley, 1984).

The model I will describe was studied in collaboration with Mary S. Riley (Riley & Smolensky, 1984) and Peter DeMarzo (1984). This model provides answers, without any symbolic manipulation of rules, to qualitative questions about the particular circuit of Figure 16. It should not be assumed that we imagine that a different harmony network like the one I will describe is created for every different circuit that is analyzed. Rather we assume that experts contain a small number of fixed networks like the one we propose, that these networks represent the effects of much cumulated experience with many different circuits, that they form the "chunks" with which the expert's *intuition* represents the circuit domain, and that complex problem solving somehow employs these networks to direct the problem solving as a whole through intuitions about chunks of the problem. At this early stage we cannot say much about the coordination of activity in complex problem solving. But we do claim that by giving an explicit example of a non-symbolic account of problem solving, our model offers insights into expertise that complement nicely those of traditional production-system models. The model also serves to render concrete many of the general features of harmony theory that have been described above.

Representational features. The first step in developing a harmony model is to select features for representing the environment. Here the environment is the set of qualitative changes in the electric circuit of Figure 16 that obey the laws of physics. What must obviously be represented are the changes in the physical components: whether R_1 goes up, goes down, or stays the same, and similarly for R_2 and the battery's voltage V_{total}. We also hypothesize that experts represent deeper features of this environment, like the current I, the voltage drops V_1 and V_2 across the two resistors, and the effective resistance R_{total} of the circuit. We claim that experts "see" these deeper features; that *perceiving* the problem of Figure 16 for experts involves filling in the deeper features just as for all sighted people—experts in vision—*perceiving* a scene involves filling in the features describing objects in three-dimensional space. Many studies of expertise in the psychological literature show that experts perceive their domain differently from novices: Their representations are much richer; they possess additional representational features that are specially developed for capturing the structure of the particular environment. (See, for example, Chase & Simon, 1973; Larkin, 1983.)

So the representational features in our model encode the qualitative changes in the seven circuit variables: R_1, R_2, R_{total}, V_1, V_2, V_{total}, and I. Our claim is that experts possess some set of features *like* these;

there are undoubtedly many other possibilities, with different sets being appropriate for modeling different experts.

Next, the three qualitative changes *up, down,* and *same* for these seven variables need to be given binary encodings. The encoding I will discuss here uses one binary variable to indicate whether there is any *change* and a second to indicate whether the change is *up*. Thus there are two binary variables, *I.c* and *I.u*, that represent the change in the current, *I*. To represent no change in *I*, the change variable *I.c* is set to -1; the value of *I.u* is, in this case, irrelevant. To represent increase or decrease of *I*, *I.c* is given the value $+1$ and *I.u* is assigned a value of $+1$ or -1, respectively. Thus the total number of representational features in the model is 14: two for each of the seven circuit variables.

Knowledge atoms. The next step in constructing a harmony model is to encode the necessary knowledge into a set of atoms, each of which encodes a subpattern of features that co-occur in the environment. The environment of idealized circuits is governed by formal laws of physics, so a specification of the knowledge required for modeling the environment is straightforward. In most real-world environments, no formal laws exist, and it is not so simple to give a priori methods for directly constructing an appropriate knowledge base. However, in such environments, the fact that harmony models encode *statistical* information rather than rules makes them much more natural candidates for viable models than rule-based systems. One way that the statistical prop-erties of the environment can be captured in the strengths of knowledge atoms is given by the learning procedure. Other methods can probably be derived for directly passing from statistics about the domain (e.g., medical statistics) to an appropriate knowledge base.

The fact that the environment of electric circuits is explicitly rule-governed makes a probabilistic model of intuition, like the model under construction, a particularly interesting theoretical contrast to the obvious rule-applying models of explicit conscious reasoning.

For our model we selected a minimal set of atoms; more realistic models of experts would probably involve additional atoms. A minimal specification of the necessary knowledge is based directly on the equations constraining the circuit: Ohm's law, Kirchoff's law, and the equation for the total resistance of two resistors in series. Each of these is an equation constraining the simultaneous change in three of the circuit variables. For each law, we created a knowledge atom for each combination of changes in the three variables that does not violate the law. These are memory traces that might be left behind after experiencing many problems in this domain, i.e., after observing many states of this

environment. It turns out that this process gives rise to 65 knowledge atoms,[18] all of which we gave strength 1.

A portion of the model is shown in Figure 17. The two atoms shown are respectively instances of Ohm's law for R_1 and of the formula for the total resistance of two resistors in series.

It can be shown that with the knowledge base I have described, whenever a completion problem posed has a unique correct answer, that answer will correspond to the state with highest harmony. This assumes that κ is set within the range determined by Theorem 1: the perfect matching limit.[19]

The parameter κ. According to the formula defining the perfect matching limit, κ must be less than 1 and greater than $1 - 2/6 = 2/3$ because the knowledge atoms are never connected to more than 6 features (two binary features for each of three variables). In the

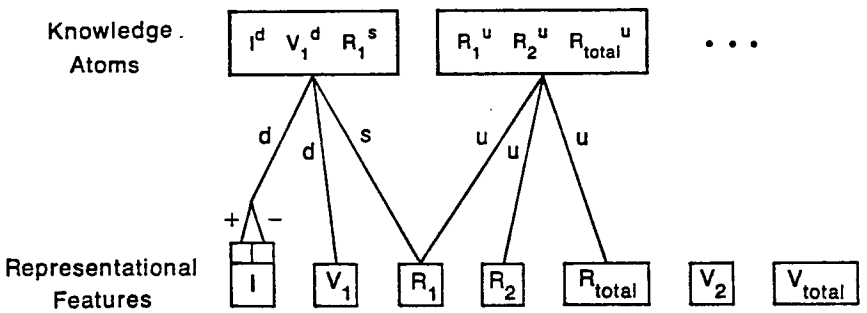

FIGURE 17. A schematic diagram of the feature nodes and two knowledge atoms of the model of circuit analysis. *u*, *d*, and *s* denote *up*, *down*, and *same*. The box labeled *I* denotes the *pair* of binary feature nodes representing *I*, and similarly for the other six circuit variables. Each connection labeled *d* denotes a *pair* of connections labeled with the binary encoding (+,−) representing *down*, and similarly for connections labeled *u* and *s*.

---

[18] Ohm's law applies three times for this circuit; once each for R_1, R_2, and R_{total}. This together with the other two laws gives five constraint equations. In each of these equations, the three variables involved can undergo 13 combinations of qualitative changes.

[19] *Proof:* The correct answer satisfies all five circuit equations, the maximum possible. Thus it exactly matches five atoms, and no possible answer can exactly match more than five atoms. In the exact matching limit, any nonexact matches cannot produce higher harmony, so the correct answer has the maximum possible harmony. If enough information is given in the problem so that there is only one correct answer, then there is only one state with this maximal harmony value.

simulations I will describe, κ was actually raised during the computation to a value of .75, as shown in Figure 18. (The model actually performs better if $\kappa = .75$ throughout: DeMarzo, 1984.)

Cooling schedule. It was not difficult to find a cooling rate that permitted the model to get the correct answer to the problem shown in Figure 16 on 28 out of 30 trials. This cooling schedule is shown in Figure 19.[20] The initial temperature (4.0) was chosen to be sufficiently high

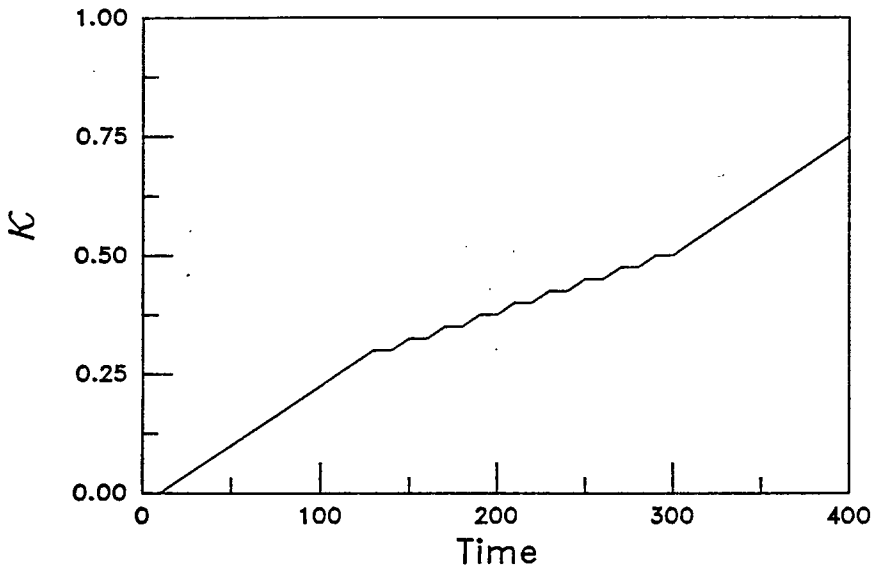

FIGURE 18. The schedule showing κ as a function of time during the computation.

[20] In the reported simulations, one node, selected randomly, was updated at a time. The computation lasted for 400 "iterations" of 100 node updates each; that is, on the average each of the 79 nodes was updated about 500 times. "Updating" a node means deciding whether to change the value of that node, regardless of whether the decision changes the value. (*Note on "psychological plausibility":* 500 updates may seem like a lot to solve such a simple problem. But I claim the model cannot be dismissed as implausible on this ground. According to current *very general* hypotheses about neural computation [see Chapter 20], each node update is a computation *comparable* to what a neuron can perform in its "cycle time" of about 10 msec. Because harmonium could actually be implemented in parallel hardware, in accordance with the realizability theorem, the 500 updates could be achieved in 500 cycles. With the cycle time of the neuron, this comes to about 5 seconds. This is clearly the correct order of magnitude for solving such problems intuitively. While it is also possible to solve such problems by firing a few symbolic productions, it is not so clear that an implementation of a production system model could be devised that would run in 500 cycles of parallel computations comparable to neural computations.)

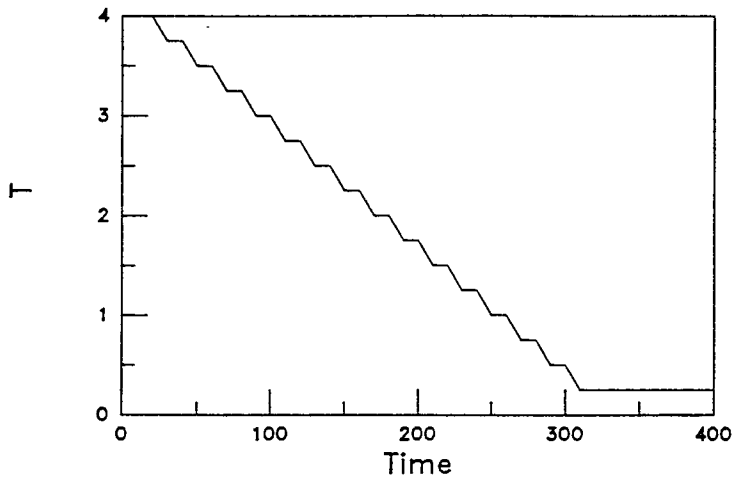

FIGURE 19. The schedule showing T as a function of time during the computation.

that nodes were flipping between their values essentially at random; the final temperature (0.25) was chosen to be sufficiently small that the representational features hardly ever flipped, so that the completion could be said to be its "final decision." Considerable computation time was probably wasted at the upper and lower ends of the cooling schedule.

The simulation. The graphical display used in the simulation provides a useful image of the computational process. On a gray background, each node was denoted by a box that was white or black depending on the current node value. Throughout the computation, the nodes encoding the given information maintain their fixed values (colors). Initially, all the atoms are black (inactive) and the unknown features are assigned random colors. When the computation starts, the temperature is high, and there is much flickering of nodes between black and white. At any moment many atoms are active. As computation proceeds and the system cools, each node flickers less and less and eventually settles into a final value.[21] The "answer" is read out by

[21] It may happen that some representation variables will be connected only to knowledge atoms that are inactive towards the end of the computation; these representation variables will continue to flicker at arbitrarily low temperatures, spending 50% of the time in each state. In fact, this happens for bits of the representation (like $R_1.u$) that encode the "direction of change" of circuit variables that are in state *no change*, indicated by – on the "presence of change" bit. These bits are ignored by the active knowledge atoms (those involving *no change* for the circuit variable) and are also ignored when we "read out" the final answer produced by the network.

decoding the features for the unknowns. Ninety-three percent of the time, the answer is correct.

The microdescription of problem solving. Since the model correctly answers physics questions, it "acts as though" it knows the symbolic rules governing electric circuits. In other words, the *competence* of the harmonium model (using Chomsky's meaning of the word) could be accurately described by symbolic inference procedures (e.g., productions) that operate on symbolic representations of the circuit equations. However the *performance* of the model (including its occasional errors) is achieved without interpreting symbolic rules.[22] In fact, the process underlying the model's performance has many characteristics that are not naturally represented by symbolic computation. The answer is computed through a series of many node updates, each of which is a *microdecision* based on formal *numerical* rules and numerical computations. These microdecisions are made many times, so that the eventual values for the different circuit variables are in an important sense being computed *in parallel. Approximate matching* is an important part of the use of the knowledge: Atoms whose feature patterns approximately match the current feature values are more likely to become active by thermal noise than atoms that are poorer matches (because poorer matches lower the harmony by a greater amount). And all the knowledge that is active at a given moment *blends* in its effects: When a given feature updates its value, its microdecision is based on the weighted sum of the recommendations from all the active atoms.

The macrodescription of problem solving. When watching the simulation, it is hard to avoid anthropomorphizing the process. Early on, when a feature node is flickering furiously, it is clear that "the system can't make up its mind about that variable yet." At some point during the computation, however, the node seems to have stopped flickering—"it's decided that the current went down." It is reasonable to say that a *macrodecision* has been made when a node stops flickering,

---

[22] The distinction between characterizing the competence and performance of dynamical systems is a common one in physics, although I know of no terminology for it. A production system expressing the circuit laws can be viewed as a *grammar for generating the high-harmony states* of the dynamical system. These laws neatly express the states into which the system will settle. However, completely different laws govern the *dynamics* through which the system enters equilibrium states. Other examples from physics of this distinction are to be found essentially everywhere. Kepler's laws, for example, neatly characterize the planetary orbits, but completely different laws, Newton's laws of motion and gravitation, describe the dynamics of planetary motion. Balmer's formula neatly characterizes the light emitted by the hydrogen atom, but utterly different laws of quantum physics describe the dynamics of the process.

although there seems to be no natural formal definition for the concept. To study the properties of macrodecisions, it is appropriate to look at how the *average values* of the stochastic node variables change during the computation. For each of the unknown variables, the node values were averaged over 30 runs of the completion problem of Figure 16, separately for each time during the computation. The resulting graphs are shown in Figure 20. The plots hover around 0 initially, indicating that values + and − are equally likely at high temperatures—lots of flickering. As the system cools, the average values of the representation variables drift toward the values they have in the correct solution to the problem ($R_{total} = up$, $I = down$, $V_1 = down$, $V_2 = up$).

Emergent seriality. To better see the macrodecisions, in Figure 21 the graphs have been superimposed and the "indecisive" band around 0 has been removed. The striking result is that out of the statistical din of parallel microdecisions emerges a *sequence* of macrodecisions.

Propagation of givens. The result is even more interesting when it is observed that in symbolic forward-chaining reasoning about this problem, the decisions are made in the order R, I, V_1, V_2. Thus not only is the *competence* of the model neatly describable symbolically, but even the *performance*, when described at the macrolevel, could be modeled by the sequential firing of productions that chain through the inferences. Of course, macrodecisions emerge first about those variables that are most directly constrained by the given inputs, but not because rules are being used that have conditions that only allow them to apply when all but one of the variables is known. Rather it is because the variables given in the input *are fixed and do not fluctuate*: They provide the information that is the most consistent over time, and therefore the knowledge consistent with the input is most consistently activated, allowing those variables involved in this knowledge to be more consistently completed than other variables. As the temperature is lowered, those variables "near" the input (with respect to the connections provided by the knowledge) stop fluctuating first, and their relative constancy of value over time makes them function somewhat like the original input to support the next wave of completion. In this sense, the stability of variables "spreads out" through the network, starting at the inputs and propagating with the help of cooling. Unlike the simple feedforward "spread of activation" through a standard activation network, this process is a spread of feedback-mediated *coherency* through a decision-making network. Like the growth of droplets or crystals, this amounts to the expansion of pockets of order into a sea of disorder.

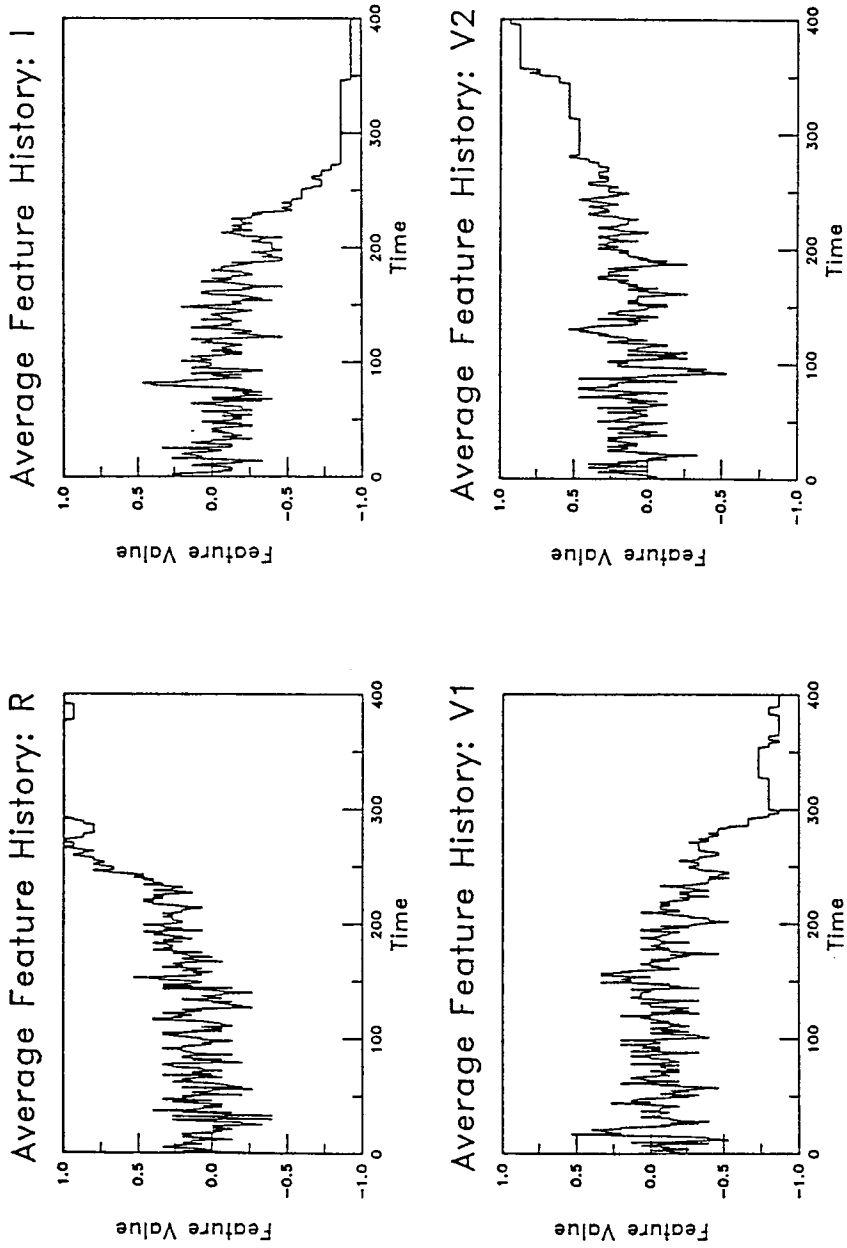

FIGURE 20. The model's hypothesized qualitative values for unknown circuit variables, averaged over 30 runs, for each iteration separately. (+ means *up*.)

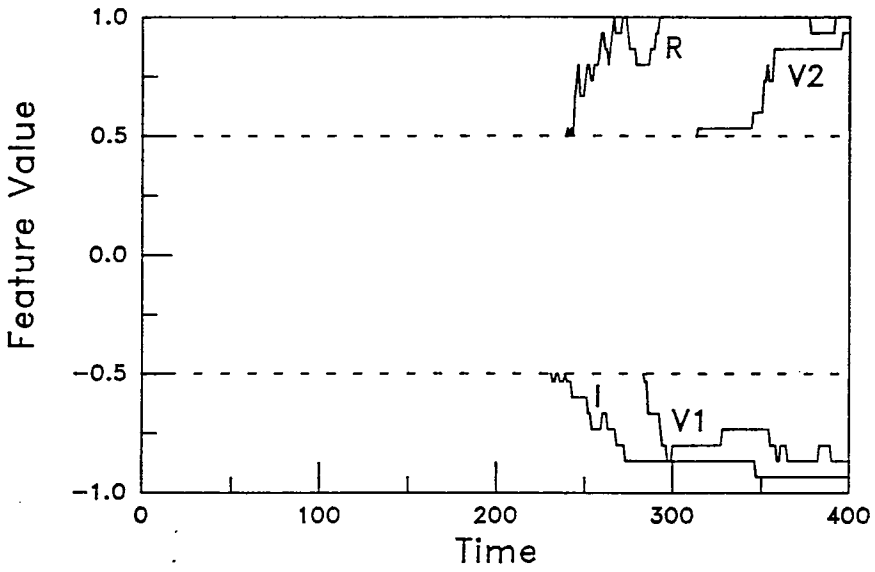

FIGURE 21. Emergent seriality: The decisions about the direction of change of the circuit variables "freeze in" in the order $R = R_{total}$, $I = I_{total}$, V_1, V_2 (R and I are very close).

Phase transition. In the previous section, a highly idealized decision-making model was seen to have a freezing temperature at which the system behavior changed from disordered (undecided) to ordered (decided). Does the same thing occur in the more complicated circuit model? As a signal for such a phase transition, physics says to look for a sharp peak in the quantity

$$C = \frac{<H^2> - <H>^2}{T^2}.$$

This is global property of the system which is proportional to the rate at which entropy—disorder—decreases as the temperature decreases; in physics, it is called the *specific heat.* If there is rapid increase in the order of the system at some temperature, the specific heat will have a peak there.

Figure 22 shows that indeed there is a rather pronounced peak. Does this macrostatistic of the system correspond to anything significant in the macrodecision process? In Figure 23, the specific heat curve is superimposed on Figure 21. The peak in the specific heat coincides remarkably with the first two, major macrodecisions about the total resistance and current.

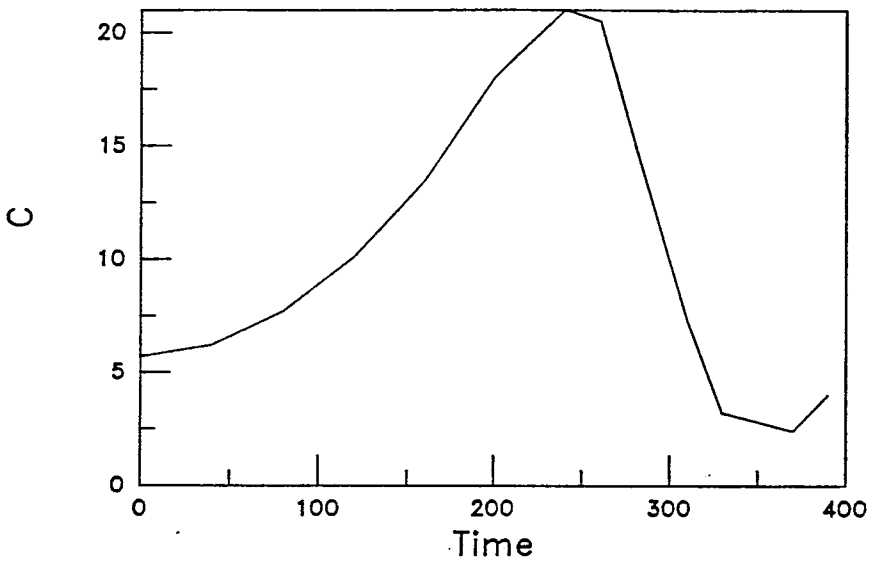

FIGURE 22. The specific heat of the circuit analysis model through the course of the computation.

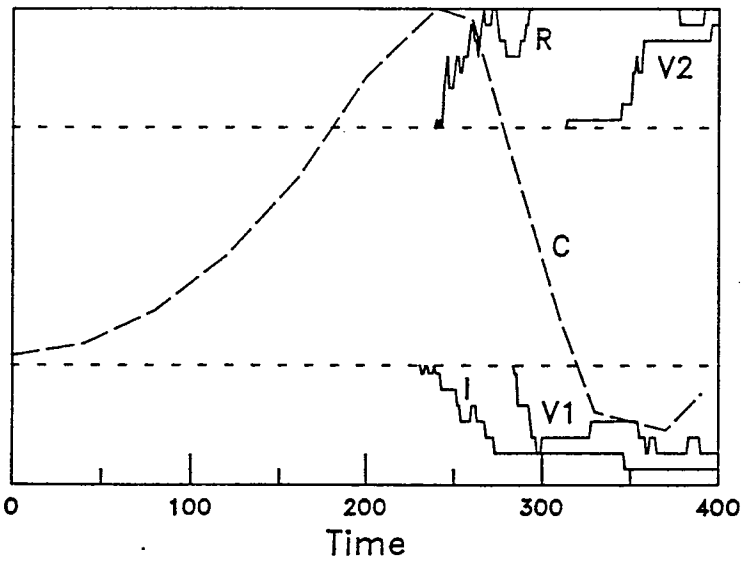

FIGURE 23. There is a peak in the specific heat at the time when the *R* and *I* decisions are being made.

MACRODESCRIPTION: PRODUCTIONS, SCHEMATA, AND EXPERTISE

Productions and Expertise

While there are similarities in the production-system account of problem solving and the macrodescription of the harmony account, there are important differences. These differences are most apparent in the accounts of how experts' knowledge is acquired and represented.

A symbolic account of expertise acquisition. A standard description within the symbolic paradigm of the acquisition of expertise is based on the idea of knowledge *compilation* (Anderson, 1982). Applied to circuit analysis, the account goes roughly like this. Novices have procedures for inspecting equations and using them to assign values to unknowns. At this stage of performance, novices consciously scan equations when solving circuit problems. As circuit problems are solved, knowledge is *proceduralized*: specialized circuit-analysis productions are stored in the knowledge base. An example of might be "IF given: R_1 and R_2 both go up, THEN conclude: R_{total} goes up" which can be abbreviated $R_1{}^u R_2{}^u \rightarrow R_{total}{}^u$. Another might be $R_{total}{}^u V_{total}{}^s \rightarrow I^d$. At this stage of performance, a series of logical steps is consciously experienced, but no equations are consciously searched. As the circuit productions are used together to solve problems, they are *composed* together (Lewis, 1978). The two productions just mentioned, for example, are composed into a single production, $R_1{}^u R_2{}^u V_{total}{}^s \rightarrow R_{total}{}^u I^d$. As the productions are composed, the conditions and actions get larger, more is inferred in each production firing, and so fewer productions need to fire to solve a given problem. Eventually, the compilation process has produced productions like $R_1{}^u R_2{}^u V_{total}{}^s \rightarrow R_{total}{}^u I^d V_1{}^d V_2{}^u$. Now we have an expert who can solve the problem in Figure 16 all at once, by firing this single production. The reason is that the knowledge base contains, prestored, a rule that says "whenever you are given this problem, give this answer."

A subsymbolic account. By contrast, the harmony theory account of the acquisition of expertise goes like this. (This account has not yet been tested with simulations.) Beginning physics students are novices in circuit analysis but experts (more or less) at symbol manipulation. Through experience with language and mathematics, they have built up—by means of the learning process referred to in the learnability theorem—a set of features and knowledge atoms for the perception and manipulation of symbols. These can be used to inspect the circuit

equations and draw inferences from them to solve circuit problems. With experience, features dedicated to the perception of circuits evolve, and knowledge atoms relating these features develop. The final network for circuit perception contains within it something like the model described in the previous section (as well as other portions for analyzing other types of simple circuits). This final network can solve the entire problem of Figure 16 in a single cooling. Thus experts perceive the solution in a single conscious step. (Although sufficiently careful perceptual experiments that probe the internal structure of the construction of the percept should reveal the kind of sequential filling-in that was displayed by the model.) Earlier networks, however, are not sufficiently well-tuned by experience; they can only solve *pieces* of the problem in a single cooling. Several coolings are necessary to solve the problem, and the answer is derived by a series of consciously experienced steps. (This gives the symbol-manipulating network a chance to participate, offering justifications of the intuited conclusions by citing circuit laws.) The number of circuit constraints that can be satisfied in parallel during a single cooling grows as the network is learned. *Productions are higher level descriptions of what input/output pairs—completions—can be reliably performed by the network in a single cooling.* Thus, in terms of their productions, novices are described by productions with simple conditions and actions, and experts are described by complex conditions and actions.

Dynamic creation of productions. The point is, however, that in the harmony theory account, *productions are just descriptive entities; they are not stored, precompiled, and fed through a formal inference engine;* rather they are *dynamically created* at the time they are needed by the appropriate collective action of the small knowledge atoms. Old patterns that have been stored through experience can be recombined in completely novel ways, giving the appearance that productions had been precompiled even though the particular condition/action pair had never before been performed. When a familiar input is changed slightly, the network can settle down in a slightly different way, flexing the usual production to meet the new situation. Knowledge is not stored in large frozen chunks; the productions are truly context sensitive. And since the productions are created on-line by combining many small pieces of stored knowledge, the set of available productions has a size that is an exponential function of the number of knowledge atoms. The exponential explosion of compiled productions is virtual, not precompiled and stored.

Contrasts with logical inference. It should be noted that the harmonium model can answer ill-posed questions just as it can answer

well-posed ones. If insufficient information is provided, there will be more than one state of highest harmony, and the model will choose one of them. It does not stop dead due to "insufficient information" for any formal inference rule to fire. If inconsistent information is given, no available state will have a harmony as high as that of the answer to a well-posed problem; nonetheless, those answers that violate as few circuit laws as possible will have the highest harmony and one of these will therefore be selected. It is not the case that "any conclusion follows from a contradiction." The mechanism that allows harmonium to solve well-posed problems allows it to find the best possible answers to ill-posed problems, with no modification whatever.

Schemata

Productions are higher level descriptions of the completion process that ignore the internal structures that bring about the input/output mapping. Schemata are higher level descriptions of chunks of the knowledge base that ignore the internal structure within the chunk. To suggest how the relation between knowledge atoms and schemata can be formalized, it is useful to begin with the idealized two-choice decision model discussed in the preceding section entitled Decision-Making and Freezing.

Two-choice model. In this model, each knowledge atom had either all + or all − connections. To form a higher level description of the knowledge, let's lump all the + atoms together into the + *schema*, and denote it with the symbol S_+. The *activation level* of this schema, $A(S_+)$, will be defined to be the average of the activations of its constituent atoms. Now let us consider all the feature nodes together as a *slot* or *variable*, s, for this schema. There are two states of the slot that occur in completions: all + and all −. We can define these to be the possible *fillers* or *values* of the slot and symbolize them by f_+ and f_-. The information in the schema S_+ is that the slot s should be filled with f_+; the proposition $s = f_+$. The "degree of truth" of this proposition, $\tau(s = f_+)$, can be defined to be the average value of all the feature nodes comprising the slot: If they are all +, this is 1 or *true*; if all − this is −1 or *false*. At intermediate points in the computation when there may be a mixture of signs on the feature nodes, the degree of truth is somewhere between 1 and −1.

Repeating the construction for the schema S_-, we end up with a higher level description of the original model depicted in Figure 24.

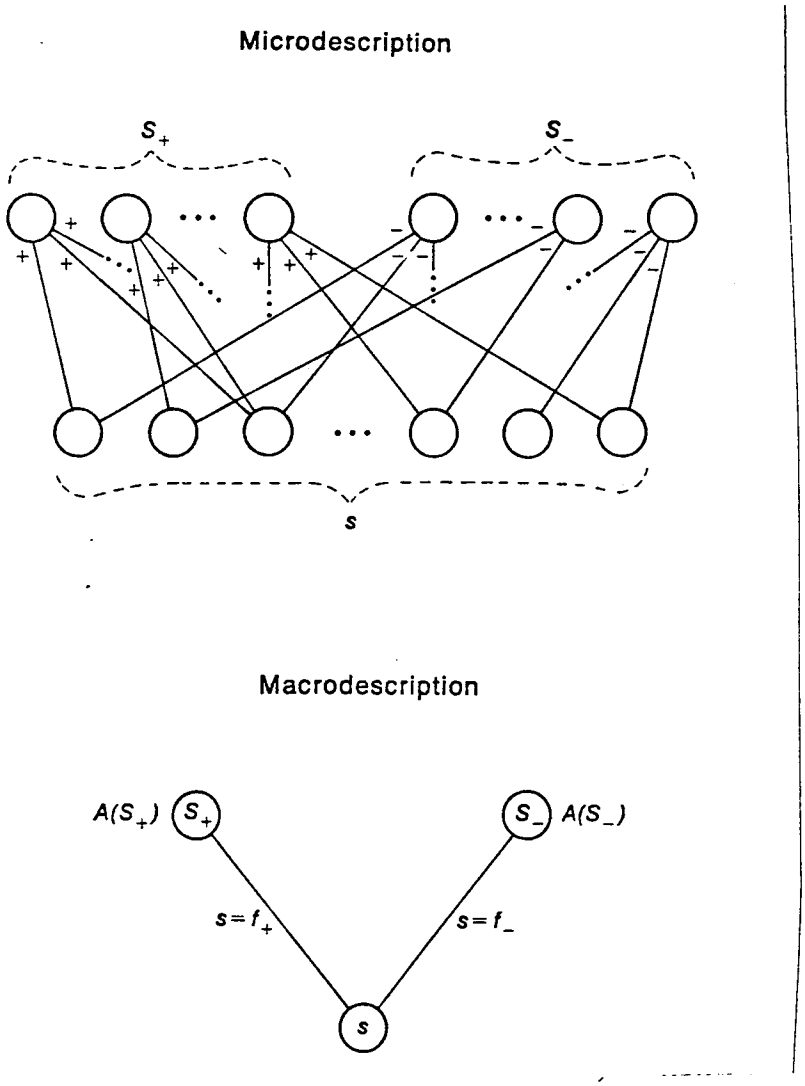

FIGURE 24. Micro- and macrodescriptions of the idealized decision model.

The interesting fact is that the harmony of any state of the original model can now be re-expressed using the higher level variables:

$$H = A(S_+)[\tau(s = f_+) - \kappa] + A(S_-)[\tau(s = f_-) - \kappa].$$

In this simple homogeneous case, the aggregate higher level variables contain sufficient information to exactly compute the harmony function.

The analysis of decision making in this model considered the limit as the number of features and atoms goes to infinity—for only in this "thermodynamic limit" can we see real phase transitions. In this limit, the set of possible values for the averages that define the aggregate variables comes closer and closer to a continuum. The central limit theorem constrains these averages to deviate less and less from their means; statistical fluctuations become less and less significant; the model's behavior becomes more and more deterministic.

Thus, just as the statistical behavior of matter disappears into the deterministic laws of thermodynamics as systems become macroscopic in size, so the statistical behavior of individual features and atoms in harmony models becomes more and more closely approximated by the higher level description in terms of schemata as the number of constituents aggregated into the schemata increases. However there are two important differences between harmony theory and statistical physics relevant here. First, the number of constituents aggregated into schemata is *nowhere near* the number—10^{23}—of particles aggregated into bulk matter. Schemata provide a useful but significantly limited description of real cognitive processing. And second, the process of aggregation in harmony theory is *much* more complex than in physics. This point can be brought out by passing from the grossly oversimplified two-choice decision model just considered to a more realistic cognitive domain.

Schemata for rooms. In a realistically complicated and large network, the schema approximation would go something like this. The knowledge atoms encode clusters of values for features that occur in the environment. Commonly recurring clusters would show up in many atoms that differ slightly from each other. (In a different language, the many exemplars of a schema would correspond to knowledge atoms that differ slightly but share many common features.) These atoms can be aggregated into a *schema*, and their average activation at any moment defines the *activation* of the schema. Now among the atoms in the cluster corresponding to a schema for a *living-room*, for example, might be a subcluster corresponding to the schema for *sofa/coffee-table*. These atoms comprise a *subschema* and the average of their activations would be the activation variable for this subschema.

The many atoms comprising the schema for *kitchen* share a set of connections to representational features relating to cooking devices. It is convenient to group together these connections into a *cooking-device slot*, $s_{cooking}$. Different atoms for different instances of *kitchen* encode various patterns of values over these representational features, corresponding to instances of *stove*, *conventional oven*, *microwave oven*, and so forth. Each of these patterns defines a possible *filler*, f_k, for the

slot. The degree of truth of a proposition like $s_{cooking} = f_i$ is the number of matches minus the number of mismatches between the pattern defining f_i and the current values over the representation nodes in the slot $s_{cooking}$, all divided by the total number of features in the slot. Now the harmony obtained by activating the schema is determined by the degrees of truth of propositions specifying the possible fillers for the slots of the schema. Just like in the simple two-decision model, the harmony function, originally expressed in terms of the microscopic variables, can be re-expressed in terms of the macroscopic variables, the activations of schemata, and slot fillers. However, since the knowledge atoms being aggregated no longer have exactly the same links to features, the new expression for H in terms of aggregate variables is only *approximately* valid. The macrodescription involves fewer variables, but the structure of these variables is more complex. The objects are becoming richer, more like the structures of symbolic computation.

This is the basic idea of the analytic program of harmony theory for relating the micro- and macro-accounts of cognition. Macroscopic variables for schemata, their activations, their slots, and propositional content are defined. The harmony function is approximately rewritten in terms of these aggregate variables, and then used to study the macroscopic theory that is determined by that new function of the new variables. This theory can be simulated, defining macroscopic models. The nature of the approximation relating the macroscopic to the microscopic models is clearly articulated, and the situations and senses in which this approximation is valid are therefore specified.

The kind of variable aggregation involved in the schema approximation is in an important respect quite unlike any done in physics. The physical systems traditionally studied by physicists have *homogeneous structure*, so aggregation is done in homogeneous ways. In cognition, the distinct roles played by different schemata mean aggregates must be specially defined. The theory of the schema limit corresponds at a very general level to the theory of the thermodynamic limit, but is rather sharply distinguished by a much greater complexity.

The Schema Approximation

In this subsection I would like to briefly discuss the schema approximation in a very general information-processing context.

In harmony theory, the cognitive system fills in missing information with reference to an internal model of the environment represented as

a probability distribution. Such a distribution of course contains potentially a phenomenal amount of information: the joint statistics of all combinations of all features used to represent the environment. How can we hope to encode such a distribution effectively? Schemata provide an answer. They comprise a way of breaking up the environment into modules—schemata—that can individually by represented as a miniprobability distribution. These minidistributions must then be folded together during processing to form an estimate of the whole distribution. To analyze a room scene, we don't need information about the joint probability of all possible features; rather, our schema for "chair" takes care of the joint probability of the features of chairs; the schema for "sofa/coffee-table" contains information about the joint probability of sofa and coffee-table features, and so on. Each schema *ignores* the features of the others, by and large.

This modularization of the encoding can reduce tremendously the amount of information the cognitive system needs to encode. If there are f binary features, the whole probability distribution requires 2^f numbers to specify. If we can break the features into s groups corresponding to schemata, each involving f/s features, then only $s2^{f/s}$ numbers are needed. This can be an enormous reduction; even with such small numbers as $f = 100$ and $s = 10$, for example, the reduction factor is $10\times 2^{-90} \approx 10^{-28}$.

The reduction in information afforded by schemata amounts to an assumption that the probability distribution representing the environment has a special, modular structure—at least, that it can be usefully so approximated. A very crude approximation would be to divide the features into disjoint groups, to separately store in schemata the probabilities of possible combinations of features within each group, and then to simply *multiply* together these probabilities to estimate the joint probability of all features. This assumes the features in the groups are completely *statistically independent*, that the values of features of a chair interact with other features of the chair but not with features of the sofa. To some extent this assumption is valid, but there clearly are limits to its validity.

A less crude approximation is to allow schemata to share features so that the shared features can be constrained simultaneously by the joint probabilities with the different sets of variables contained in the different schemata to which it relates. Now we are in the situation modeled by harmony theory. A representational feature node can be attached to many knowledge atoms and thereby participate in many schemata. The distribution $e^{H/T}$ manages to combine into a single probability distribution all the separate but interacting distributions corresponding to the separate schemata. Although the situation is not

as simple as the case of nonoverlapping schemata and completely independent subdistributions, the informational savings is still there. The trick is to isolate groups of environmental features which each comprise a small fraction of the whole feature set, to use these groups to define more abstract features, and record the probability distributions using these features. The groups must be selected to capture the most important interrelationships in the environment. This is the problem of constructing new features. The last section offers a few comments on this most important issue.

LEARNING NEW REPRESENTATIONS

The Learning Procedure and Abstract Features

Throughout this chapter I have considered cognitive systems that represent states of their environment using features that were established prior to our investigation, either through programming by the modeler, or evolution, or learning. In this section I would like to make a few comments about this last possibility, the establishment of features through learning.

Throughout this chapter I have emphasized that the features in harmony models represent the environment at all levels of abstractness. In the preceding account of how expertise in circuit analysis is acquired, it was stated that through experience, experts evolve abstract features for representing the domain. So the basic notion is that the cognitive system comes into existence with a set of *exogenous features* whose values are determined completely by the state of the external environment, whenever the environment is being observed. Other *endogenous features* evolve, through a process now to be described, through experience, from an initial state of meaninglessness to a final state of abstract meaning. Endogenous features always get their values through internal completion, and never directly from the external environment.[23]

As a specific example, consider the network of Figure 9, which is repeated as Figure 25. In this network, features of several levels of

[23] In Chapter 7, Hinton and Sejnowski use the terms *visible* and *hidden* units. The former correspond to the exogenous feature nodes, while the latter encompass *both* the endogenous feature nodes and the knowledge atoms.

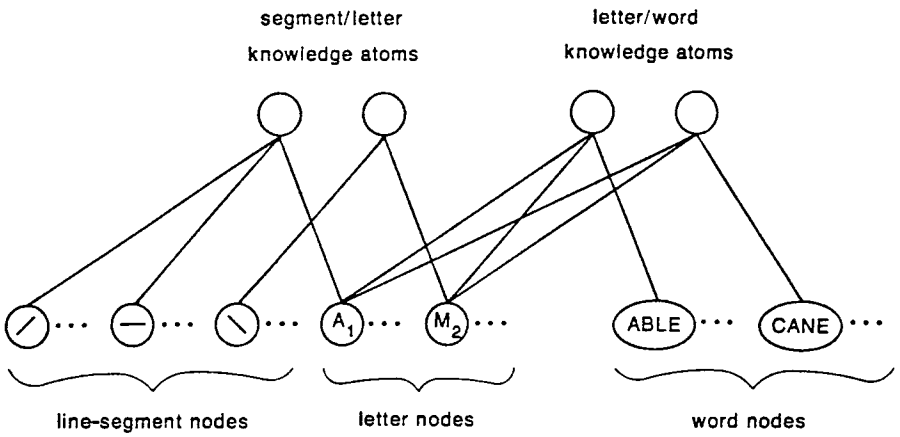

FIGURE 25. A network representing words at several levels of abstractness.

abstractness are used to represent words. Here is a *hypothetical* account of how such a network could be learned.[24]

The features representing the line segments are taken to be the exogenous features given a priori. This network comes into existence with these line-segment nodes, together with extra endogenous feature nodes which, through experience, will become the letter and word nodes.

As before, the cognitive system is assumed to come into existence with a set of knowledge atoms whose strengths will be adjusted to match the environment. Some of these atoms have connections only to exogenous features, some only to endogenous features, and some to both types of features.

The environment (in this case, a set of words) is observed. Each time a word is presented, the appropriate values for the line-segment nodes are set. The current atom strengths are used to complete the input, through the cooling procedure discussed above. The endogenous features are thus assigned values for the particular input. Initially,

[24] The issue of selecting patterns on exogenous features for use in defining endogenous features—including the word domain—is discussed in Smolensky (1983). To map the terminology of that paper on to that of this chapter, replace *schemas* by *knowledge atoms* and *beliefs* by *feature values*. That paper offers an alternative use of the harmony concept in learning. Rather than specifying a learning process, it specifies an optimality condition on the atom strengths: They should maximize the total harmony associated with interpreting all environmental stimuli. This condition is related, but not equivalent, to information-theoretic conditions on the strengths.

when the atoms' strengths have received little environmental tuning, the values assigned to the endogenous features will be highly random. Nonetheless, after the input has been completed, learning occurs: The strengths of atoms that match the feature nodes are all increased by $\Delta\sigma$.

Intermixed with this incrementing of strengths during environmental observation is a process of decrementing strengths during environmental simulation. Thus the learning process is exactly like the one referred to in the learnability theorem, except that now, during observation, not all the features are set by the environment; the endogenous features must be filled in by completion.

Initially, the values of the endogenous features are random. But as learning occurs, correlations between recurring patterns in the exogenous features and the random endogenous features will be amplified by the strengthening of atoms that encode those correlations. An endogenous feature by chance tends to be + when patterns of line segments defining the letter *A* are present and so leads to strengthening of atoms relating it to those patterns; it gradually comes to represent *A*. In this way, self-organization of the endogenous features can potentially lead them to acquire meaning.

The learnability theorem states that when no endogenous features are present, this learning process will produce strengths that optimally encode the environmental regularities, in the sense that the completions they give rise to are precisely the maximum-likelihood completions of the estimated environmental probability distribution with maximal missing information that is consistent with observable statistics. At present there is no comparable theorem that guarantees that in the presence of endogenous features this learning procedure will produce strengths with a corresponding optimality characterization.[25]

Among the most important future developments of the theory is the study of self-organization of endogenous features. These developments include a possible extension of the learnability theorem to include endogenous features as well as computer simulations of the learning procedure in specific environments.

---

[25] In Chapter 7, Hinton and Sejnowski use a different but related optimality condition. They use a function G which measures the information-theoretic difference between the true environmental probability distribution and the estimated distribution e^H. For the case of no endogenous features, the following is true (see Theorem 4 of the Appendix). The strengths that correspond to the maximal-missing-information distribution consistent with observable statistics are the same as the strengths that minimize G. That the estimated distribution is of the form e^H must be *assumed* a priori in using the minimal-G criterion; it is *entailed* by the maximal-missing-information criterion.

Learning in the Symbolic and Subsymbolic Paradigms

Nowhere is the contrast between the symbolic and subsymbolic approaches to cognition more dramatic than in learning. Learning a new concept in the symbolic approach entails creating something like a new schema. Because schemata are such large and complex knowledge structures, developing automatic procedures for generating them in original and flexible ways is extremely difficult.

In the subsymbolic account, by contrast, a new schema comes into being gradually, as the strengths of atoms slowly shifts in response to environmental observation, and new groups of coherent atoms slowly gain important influence in the processing. During learning, there need never be any decision that "now is the time to create and store a new schema." Or rather, if such a decision is made, it is by the modeler *observing* the evolving cognitive system and not by the system itself.

Similarly there is never a time when the cognitive system decides "now is the time to assign this meaning to this endogenous feature." Rather, the strengths of all the atoms that connect to the given endogenous feature slowly shift, and with it the "meaning" of the feature. Eventually, the atoms that emerge with dominant strength may create a network like that of Figure 25, and the modeler observing the system may say "this feature means the letter *A* and this feature the word *ABLE*." Then again, some completely different representation may emerge.

The reason that learning procedures can be derived for subsymbolic systems, and their properties mathematically analyzed, is that in these systems knowledge representations are extremely impoverished. It is for this same reason that they are so hard for us to program. It is therefore in the domain of learning, more than any other, that the potential seems greatest for the subsymbolic paradigm to offer new insights into cognition. Harmony theory has been motivated by the goal of establishing a subsymbolic computational environment where the mechanisms for *using* knowledge are simultaneously sufficiently powerful and analytically tractable to facilitate—rather than hinder—the study of learning.

CONCLUSIONS

In this chapter I have described the foundations of harmony theory, a formal subsymbolic framework for performing an important class of generalized perceptual computations: the completion of partial

descriptions of static states of an environment. In harmony theory, knowledge is encoded as constraints among a set of well-tuned perceptual features. These constraints are numerical and are imbedded in an extremely powerful parallel constraint satisfaction machine: an informal inference engine. The constraints and features evolve gradually through experience. The numerical processing mechanisms implementing both performance and learning are derived top-down from mathematical principles. When the computation is described on an aggregate or macrolevel, qualitatively new features emerge (such as seriality). The *competence* of models in this framework can sometimes be neatly expressed by symbolic rules, but their *performance* is never achieved by explicitly storing these rules and passing them through a symbolic interpreter.

In harmony theory, the concept of self-consistency plays the leading role. The theory extends the relationship that Shannon exploited between information and physical entropy: Computational self-consistency is related to physical energy, and computational randomness to physical temperature. The centrality of the consistency or harmony function mirrors that of the energy or Hamiltonian function in statistical physics. Insights from statistical physics, adapted to the cognitive systems of harmony theory, can be exploited to relate the micro- and macrolevel accounts of the computation. Theoretical concepts, theorems, and computational techniques are being pursued, towards the ultimate goal of a subsymbolic formulation of the theory of information processing.

ACKNOWLEDGMENTS

The framework presented in this chapter grew out of an attempt to formalize approaches to understanding cognition that I have learned from Dave Rumelhart, Doug Hofstadter, and Geoff Hinton. I thank them for sharing their insights with me over several years. Thanks too to Steve Greenspan, Jay McClelland, Mary Riley, Gerhard Dirlich, Francis Crick, and especially Stu Geman for very instructive conversations. Peter DeMarzo has made important contributions to the theory and I have benefited greatly from working with him. I would like to thank the members of the UCSD Cognitive Science Lab and particularly the Parallel Distributed Processing research group for all their help and support. Special thanks go to Judith Stewart, Dan Rabin, Don Gentner, Mike Mozer, Rutie Kimchi, Don Norman, and Sondra Buffett. Thanks to Eileen Conway and Mark Wallen for excellent graphics and computer support. The work was supported by the System

Development Foundation, the Alfred P. Sloan Foundation, National Institute of Mental Health Grant PHS MH 14268 to the Center for Human Information Processing, and Personnel and Training Research Programs of the Office of Naval Research Contract N00014-79-C-0323, NR 667-437.

APPENDIX: FORMAL PRESENTATION OF THE THEOREMS

Formal relationships between parallel (or neural) computation and statistical mechanics have been exploited by several researchers. Three research groups in particular have been in rather close contact since their initially independent development of closely related ideas. These groups use names for their research which reflect the independent perspectives that they maintain: the *Boltzmann machine* (Ackley, Hinton, & Sejnowski, 1985; Fahlman, Hinton, & Sejnowski, 1983; Hinton & Sejnowski, 1983a, 1983b; Chapter 7), the *Gibbs sampler* (Geman & Geman, 1984), and *harmony theory* (Smolensky, 1983, 1984; Smolensky & Riley, 1984). In this appendix, all results are presented from the perspective of harmony theory, but ideas from the other groups have been incorporated and are so referenced.[26]

Because the ideas have been informally motivated and pursued at some length in the text, this appendix is deliberately formal and concise. The proofs are presented in the final section. In making the formal presentation properly self-contained, a certain degree of redundancy with the text is necessarily incurred; this is an inevitable consequence of presenting the theory at three levels of formality within a single, linearly ordered document.

Preliminary Definitions

Overview of the definitions. The basic theoretical framework is schematically represented in Figure 26. There is an external environment with structure that allows prediction of which events are more likely than others. This environment is passed through transducers to become represented internally in the *exogenous features* of a representational space. (Depending on the application, the transducers might include considerable perceptual and cognitive processing, so that the exogenous features might in fact be quite high level; they are just unanalyzed at the level of the particular model.) The features in the

[26] Hofstadter (1983) uses the idea of computational temperature in a heuristic rather than formal way to modulate the parallel symbolic processing in an AI system for doing anagrams. His insights into relationships between statistical mechanics and cognition were inspirational for the development of harmony theory (see Hofstadter, 1985, pp. 654-665).

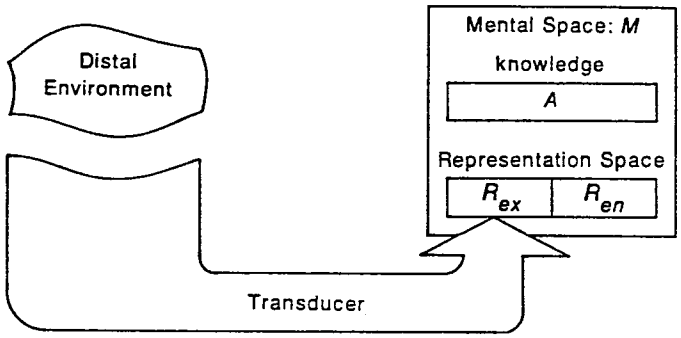

FIGURE 26. A schematic representation of the theoretical framework.

representation are taken to be binary. The prediction problem is to take some features of an environmental state as input and make best guesses about the unknown features. This amounts to extrapolating from some observed statistics of the environment to an entire probability distribution over all possible feature combinations. This extrapolation proceeds by constructing the distribution that adds minimal information (in Shannon's sense) to what is observed.

Notation. $\mathbf{B} = \{-1, +1\}$, the default *binary values*. $\mathbf{R}$ = the real numbers. $X^n = X \times X \times \cdots \times X$ (n times), where $\times$ is the cartesian product. If $\mathbf{x}, \mathbf{y} \in X^n$, then $\mathbf{x} \cdot \mathbf{y} = \sum_{m=1}^{n} x_m y_m$ and $|\mathbf{x}| = \sum_{m=1}^{n} |x_m|$. 2^X is the set of all subsets of X. $|X|$ is the number of elements of X. $\mathbf{B}^n$ is called a *binary hypercube*. The ith *coordinate function* of $\mathbf{B}^n$ ($i = 1, \ldots, n$) gives for any point (i.e., vector) in $\mathbf{B}^n$ its ith $\mathbf{B}$-valued coordinate (i.e., component).

Def. A *distal environment* $E_{distal} = (E, P)$ is a set E of *environmental events* and a probability distribution P on E.

Def. A *representational space* R is a cartesian product $R_{ex} \times R_{en}$ of two binary hypercubes. Each of the N (N_{ex}; N_{en}) binary-valued coordinate functions r_i of R (R_{ex}; R_{en}) is called an *(exogenous; endogenous) feature.*

Def. A *transduction map* T from an environment E_{distal} to a representational space $R = R_{ex} \times R_{en}$ is a map $T: E \rightarrow R_{ex}$. T induces a probability distribution p on R_{ex}: $p = P \circ T^{-1}$. This distribution is the *(proximal) environment.*

Def. Let R be a representational space. Associated with this space is the *input space* $I = \{-1, 0, +1\}^{N_{ex}}$.

Def. A point $\mathbf{r}$ in R is called a *completion* of a point ι in I if every nonzero feature of ι agrees with the corresponding feature of $\mathbf{r}$. This relationship will be designated $\mathbf{r} \supset \iota$. A *completion function* c is a map from I to 2^R (the subsets of R) for which $\mathbf{r} \in c(\iota)$ implies $\mathbf{r} \supset \iota$. The features of ι with value 0 are the "unknowns" that must be filled in by the completion function.

Def. Let p be a probability distribution on a space $X = R_{ex} \times A$. The *maximum-likelihood completion function* determined by p, $c_p: I \rightarrow 2^R$, is defined by

$$c(\iota) = \{\, \mathbf{r} \in R \mid \text{for some } \mathbf{a} \in A, \text{ and all } (\mathbf{a}', \mathbf{r}') \in R \times A \text{ such that } \mathbf{r}' \supset \iota\colon\ p(\mathbf{r}, \mathbf{a}) \geqslant p(\mathbf{r}', \mathbf{a}') \,\}$$

(A will be either empty or the set of possible knowledge atom activation vectors.)

Def. A *basic event* α has the form

$$\alpha\colon\ [r_{i_1} = b_1]\ \&\ [r_{i_2} = b_2]\ \&\ \cdots\ \&\ [r_{i_\beta} = b_\beta]$$

where $\{r_{i_1}, r_{i_2}, \ldots, r_{i_\beta}\}$ is a collection of exogenous features and $(b_1, b_2, \ldots, b_\beta) \in \mathbf{B}^\beta$. α can be characterized by the function $\chi_\alpha: R \rightarrow \{0,1\}$ defined by

$$\chi_\alpha(\mathbf{r}) = \prod_{\mu=1}^{\beta} \tfrac{1}{2}\left| r_{i_\mu}(\mathbf{r}) + b_\mu \right|$$

which is 1 if the features all have the correct values, and 0 otherwise. A convenient specification of α is as the *knowledge vector*

$$\mathbf{k}_\alpha = (0, 0, \ldots, 0, b_{i_1}, 0, \ldots, 0, b_{i_2}, 0, \ldots, 0, b_{i_\beta}, 0, \ldots, 0) \in \{-1, 0, +1\}^N$$

in which the i_μth element is b_μ and the remaining elements are all zero.

Def. A set O of *observables* is a collection of basic events.

Def. Let p be an environment and O be a set of observables. The *observable statistics* of p is the set of probabilities of all the events in O: $\{p(\alpha)\}_{\alpha \in O}$.

Def. The *entropy* (or the *missing information*; Shannon, 1948/1963) of a probability distribution p on a finite space X is

$$S(p) = -\sum_{x \in X} p(x) \ln p(x).$$

Def. The *maximum entropy estimate* $\pi_{p,O}$ of environment p with observables O is the probability distribution with maximal entropy that possesses the same observable statistics as p.

This concludes the preliminary definitions. The distal environment and transducers will play no further role in the development. They were introduced to acknowledge the important conceptual role they play: the root of all the other definitions. A truly satisfactory theory would probably include analysis of the structure of distal environments and the transformations on that structure induced by adequate transduction maps. Endogenous features will also play no further role: Henceforth, R_{en} is taken to be empty. It is an open question how to incorporate the endogenous variables into the following results. They were introduced to acknowledge the important conceptual role they must play in the future development of the theory.

Cognitive Systems and the Harmony Function H

Def. A *cognitive system* is a quintuple (R, p, O, π, c) where:

- R is a representational space,
- p is an environment,
- O is a set of statistical observables,
- π is the maximum-entropy estimate $\pi_{p,O}$ of environment p with observables O,
- c is the maximum-likelihood completion function determined by π.

Def. Let X be a finite space and $V: X \to \mathbf{R}$. The *Gibbs distribution* determined by V is

$$p_V(x) = Z^{-1} e^{V(x)}$$

where Z is the normalization constant:

$$Z = \sum_{x \in X} e^{V(x)}.$$

Theorem 1: Competence. A: The distribution π of the cognitive system (R, p, O, π, c) is the Gibbs distribution p_U determined by the function

$$U(\mathbf{r}) = \sum_{\alpha \in O} \lambda_\alpha \chi_\alpha(\mathbf{r})$$

for suitable parameters $\boldsymbol{\lambda} = \{\lambda_\alpha\}_{\alpha \in O}$ (S. Geman, personal communication, 1984). *B*: The completion function c is the maximum-likelihood completion function c_{p_H} of the Gibbs distribution p_H, where $H: M \rightarrow \mathbf{R}$, $M = R \times A$, $A = \{0,1\}^{|O|}$, is defined by

$$H(\mathbf{r}, \mathbf{a}) = \sum_{\alpha \in O} \sigma_\alpha a_\alpha h(\mathbf{r}, \mathbf{k}_\alpha)$$

and

$$h(\mathbf{r}, \mathbf{k}_\alpha) = \mathbf{r} \cdot \mathbf{k}_\alpha / |\mathbf{k}_\alpha| - \kappa$$

for suitable parameters $\boldsymbol{\sigma} = \{\sigma_\alpha\}_{\alpha \in O}$ and for κ sufficiently close to 1:

$$1 > \kappa > 1 - 2/\left[\max_{\alpha \in O} |\mathbf{k}_\alpha|\right].$$

Theorem 2 will describe how the variables $\mathbf{a} = \{a_\alpha\}_{\alpha \in O}$ can be used to actually compute the completion function. Theorem 3 will describe how the parameters $\boldsymbol{\sigma}$ can be learned through experience in the environment. Together, these theorems motivate the following interpretation.

Terminology. The triple $(\mathbf{k}_\alpha, \sigma_\alpha, a_\alpha)$ defines the *knowledge atom* or *memory trace* α. The vector $\mathbf{k}_\alpha$ is called *the knowledge vector of atom* α. The knowledge vector is an unchanging aspect of the atom. The real number σ_α is called *the strength of atom* α. This strength changes with experience in the environment. The $\{0,1\}$ variable a_α is called *the activation of atom* α. The activation of an atom changes during each computation of the completion function. The set $\mathbf{K} = \{(\mathbf{k}_\alpha, \sigma_\alpha)\}_{\alpha \in O}$ is the *long-term memory state* or *knowledge base* of the cognitive system. The vector $\mathbf{a}$ of knowledge atom activations $\{a_\alpha\}_{\alpha \in O}$ is the *working-memory state.* The value $h(\mathbf{r}, \mathbf{k}_\alpha)$ is a measure of the *consistency* between the representation vector $\mathbf{r}$ and the knowledge vector of atom α; it is the potential contribution (per unit strength) of atom α to H. The value $H(\mathbf{r}, \mathbf{a})$ is a measure of the overall consistency between the entire vector $\mathbf{a}$ of knowledge atom activations and the representation $\mathbf{r}$, relative to the knowledge base $\mathbf{K}$. Through $\mathbf{K}$, H internalizes within

the cognitive system some of the statistical regularities of the environment. Viewing the completion of an input ι as an inference process, we can say that H allows the system to distinguish which patterns of features $\mathbf{r}$ are more *self-consistent* than others, as far as the environmental regularities are concerned. This is why H is called the *harmony function*.

Def. The cognitive system determined by a harmony function H can be represented by a *graph* which will shortly be interpreted as a network of stochastic parallel processors (see Figure 27). For each coordinate of the cognitive system's *mental space* M, that is, for each feature r_i and each atom α, there is a node. These nodes carry binary values; the node for feature r_i carries the value of $r_i \in \{+1, -1\}$, while the node for atom α carries the activation value $a_\alpha \in \{1,0\}$. If the value of $\mathbf{k}_\alpha$ for a feature r_i is $+1$ or -1, there is a link with the corresponding ± 1 label joining the nodes for a_α and r_i. Finally, each node α is labeled by its strength, σ_α. The graphs of harmony networks are *two-color*; if feature nodes are assigned one color and atom nodes another, all links go between nodes of different colors. This will turn out to permit a high degree of parallelism in the processing network.

Retrieving Information From H: Performance

Def. Let $\{p_t\}_{t=0}^{\infty}$ be a sequence of probability distributions on a binary cube $X = \mathbf{B}^n$. The paths of the *(one-variable heat bath) stochastic process* $\boldsymbol{x}$ determined by $\{p_t\}$ is defined by the following procedure. At time $t = 0$, $\boldsymbol{x}$ occupies some state $\boldsymbol{x}(0) = \mathbf{x} \in X$, described by

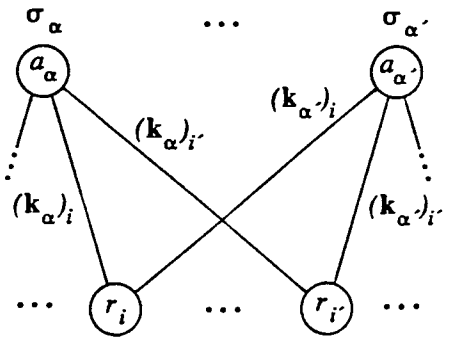

FIGURE 27. A harmony network: The graph associated with a harmony function.

some arbitrary *initial distribution*, $\mathrm{pr}(x(0) = \mathbf{x})$. Given the initial state $\mathbf{x}$, the new state at Time 1, $x(1)$, is constructed as follows. One of the n coordinates of M is selected (with uniform distribution) for *updating*. All the other $n-1$ coordinates of $x(1)$ will be the same as those of $x(0) = \mathbf{x}$. The updated coordinate can retain its previous value, leading to $x(1) = \mathbf{x}$, or it can flip to the other binary value, leading to a new state that will be denoted $\mathbf{x}'$. The selection of the value of the updated coordinate for $x(1)$ is stochastically chosen according to the likelihood ratio:

$$\frac{\mathrm{pr}(x(1) = \mathbf{x}')}{\mathrm{pr}(x(1) = \mathbf{x})} = \frac{p_0(\mathbf{x}')}{p_0(\mathbf{x})}$$

(where p_0 is the probability distribution for $t = 0$ in the given sequence $\{p_t\}_{t=0}^{\infty}$). This process—randomly select a coordinate to update and stochastically select a binary value for that coordinate—is iterated indefinitely, producing states $x(t)$ for all times $t = 1, 2, \ldots$. At each time t, the likelihood ratio of values for the stochastic choice is determined by the distribution p_t.

Def. Let p be a probability distribution. Define the one-parameter family of distributions p_T by

$$p_T = N_T^{-1} p^{1/T}$$

where the normalization constants are

$$N_T = \sum_{\mathbf{x} \in X} p(\mathbf{x})^{1/T}.$$

T is called the *temperature parameter*. An *annealing schedule* $\mathbf{T}$ is a sequence of positive values $\{T_t\}_{t=0}^{\infty}$ that converge to zero. The *annealing process* determined by p and $\mathbf{T}$ is the heat bath stochastic process determined by the sequence of distributions, p_{T_t}. If p is the Gibbs distribution determined by V, then

$$p_T(\mathbf{x}) = Z_T^{-1} e^{V(\mathbf{x})/T}$$

where

$$Z_T = \sum_{\mathbf{x} \in X} e^{V(\mathbf{x})/T}.$$

This is the same (except for the sign of the exponent) as the relationship that holds in classical statistical mechanics between the probability $p(\mathbf{x})$ of a microscopic state $\mathbf{x}$, its energy $V(\mathbf{x})$, and the temperature T. This is the basis for the names "temperature" and "annealing schedule." In the annealing process for the Gibbs distribution p_H of Theorem 1 on

the space M, the graph of the harmony network has the following significance. The updating of a coordinate can be conceptualized as being performed by a processor at the corresponding node in the graph. To make its stochastic choice with the proper probabilities, a node updating at time t must compute the ratio

$$\frac{p_{T_t}(\mathbf{x}')}{p_{T_t}(\mathbf{x})} = e^{[H(\mathbf{x}') - H(\mathbf{x})]/T_t}.$$

The exponent is the difference in harmony between the two choices of value for the updating node, divided by the current computational temperature. By examining the definitions of the harmony function and its graph, this difference is easily seen to depend only on the values of nodes connected to the updating node. Suppose at times t and $t+1$ two nodes in a harmony network are updated. If these nodes are not connected, then the computation of the second node is not affected by the outcome of the first: They are statistically independent. These computations can be performed *in parallel* without changing the statistics of the outcomes (assuming the computational temperature to be the same at t and $t+1$). Because the graph of harmony networks is two-color, this means there is another stochastic process that can be used without violating the validity of the upcoming Theorem 2.[27] *All the nodes of one color can update in parallel.* To pass from $\mathbf{x}(t)$ to $\mathbf{x}(t+1)$, all the nodes of one color update in parallel; then to pass from $\mathbf{x}(t+1)$ to $\mathbf{x}(t+2)$, all the nodes of the other color update in parallel. In twice the time it takes a processor to perform an update, plus twice the time required to pass new values along the links, a cycle is completed in which an entirely new state (potentially different in all $N + |O|$ coordinates) is computed.

> *Theorem 2: Realizability.* *A*: The heat bath stochastic process determined by p_U converges, for any initial distribution, to the distribution π of the cognitive system (R, p, O, π, c) [Metropolis et al., 1953]. *B*: The annealing process determined by p_H converges, for any initial distribution, to the completion function of the cognitive system, for any annealing schedule that approaches zero sufficiently slowly (Geman & Geman, 1984).

Part A of this theorem means the following. Suppose an input ι is given. Those features specified in ι to have values $+1$ or -1 are

[27] This is an important respect in which harmony networks differ from the arbitrary networks allowed in the Boltzmann machine.

assigned their values, which are thereafter fixed. The remaining features are assigned random initial values; these will change through the stochastic process. Now we begin the stochastic process determined by p_U. (The state space X is now R, and the same distribution p_U is used for all times.) The nonfixed variables flip back and forth between their binary values. As time progresses, the probability of finding the system in any state $\mathbf{r} \supset \iota$ approaches the maximum-entropy estimate $\pi(\mathbf{r})$ (conditioned on ι, so that only completions of ι have nonzero probability). The meaning of Part B of Theorem 1 is this: As in Part A, we fix the features specified in the input ι and start the other features off with random values. The activation variables are assigned initial values, say, of 0. We start the annealing process determined by p_H. (The state space X is now $M = R \times A$.) The unfixed features and all the activations flip between their values. The temperature drops according to the annealing schedule. As time progresses, the probability of finding the system in a state other than a maximum-likelihood completion of ι goes to zero. (If there are multiple maximum-likelihood completions, these completions become equally likely as time progresses.)

Storing Information in H: Learning

Def. (After Hinton & Sejnowski, 1983a.) Let (R, p, O, π, c) be a cognitive system. The *trace learning procedure* is defined iteratively as follows. Initially, let $\lambda_\alpha = 0$ for all $\alpha \in O$. Present the system with a sample of states, $\mathbf{r}$, drawn from the environmental distribution, p (*environmental observation*). Now store an *increment* for each λ_α equal to the mean of $\chi_\alpha(\mathbf{r})$ in this sample. Next, use the current $\boldsymbol{\lambda}$ to define U as in Theorem 1 and use the stochastic process determined by p_U to generate a sample of values of $\mathbf{r}$ from the distribution p_U, following Theorem 2 (*environmental simulation*). Now store a *decrement* for each λ_α equal to the mean of $\chi_\alpha(\mathbf{r})$ in this sample. Finally, change each λ_α by the stored increment minus the decrement. Repeat this observe-environment/simulate-environment/modify-$\boldsymbol{\lambda}$ cycle. Throughout the learning, define

$$\sigma_\alpha = \frac{\lambda_\alpha}{1 - \kappa}.$$

For small $\Delta\lambda$, a good approximate way to implement this procedure is to alternately observe and simulate the environment in equal

proportions, and to increment (respectively, decrement) λ_α by $\Delta\lambda$ each time the feature pattern defining α appears during observation (respectively, simulation). It is in this sense that σ_α is the strength of the memory trace for the feature pattern $\mathbf{k}_\alpha$ defining α. Note that in learning, equilibrium is established when the frequency of occurrence of each pattern $\mathbf{k}_\alpha$ during simulation equals that during observation (i.e., λ_α has no *net* change).

> *Theorem 3: Learnability.* Suppose all knowledge atoms are independent. Then if sufficient sampling is done in the trace learning procedure to produce accurate estimates of the observable statistics, $\boldsymbol{\lambda}$ and $\boldsymbol{\sigma}$ will converge to the values required by Theorem 1.

Independence of the knowledge atoms means that the functions $\{\chi_\alpha\}_{\alpha \in O}$ are linearly independent. This means no two atoms can have exactly the same knowledge vector. It also means no knowledge atom can be simply the "or" of some other atoms: for example, the atom with knowledge vector $+0$ is the "or" of the atoms $++$ and $+-$, and so is not independent of them. (Indeed, $\chi_{+0} = \chi_{++} + \chi_{+-}$.) The sampling condition of this theorem indicates the tradeoff between learning speed and performance accuracy. By adding higher order statistics to O (longer patterns), we can make π a more accurate representation of p and thereby increase performance accuracy, but then learning will require greater sampling of the environment.

Second-Order Observables and the Boltzmann Machine

Consider the special case in which the observables O each involve no more than two features. The largest independent set of such observables is the set of all observables either of the form

$$\alpha_i\colon\ [r_i = +]$$

or the form

$$\alpha_{ij}\colon\ [r_i = +]\ \&\ [r_j = +]$$

with $i<j$, i.e.,

$$\alpha_{ij}\colon\ \alpha_i\ \&\ \alpha_j.$$

To see that the other first- or second-order observations are not independent of these, consider a particular pair of features r_i and r_j, and let

$$\chi_{b_1 b_2} = \chi_{[r_i = b_1] \& [r_j = b_2]}$$

and

$$\chi_{b_1 0} = \chi_{[r_i = b_1]}$$

$$\chi_{0 b_2} = \chi_{[r_j = b_2]}.$$

Then notice:

$$\chi_{+-} = \chi_{+0} - \chi_{++}$$

$$\chi_{-0} = 1 - \chi_{+0}$$

$$\chi_{--} = 1 - \chi_{++} - \chi_{+-} - \chi_{-+}$$

$$= 1 - \chi_{++} - [\chi_{+0} - \chi_{++}] - [\chi_{0+} - \chi_{++}].$$

Thus, the χ-functions for all first- and second-order observations can be linearly generated from the set

$$O = \{\chi_{ij}\}_{i<j} \cup \{\chi_i\}_i$$

which will now be taken to be the set of observables. I will abbreviate $\lambda_{\alpha_{ij}}$ as λ_{ij} and λ_{α_i} as λ_i. Next, consider the U function for this set, O:

$$U = \sum_{\alpha \in O} \lambda_\alpha \chi_\alpha = \sum_{i<j} \lambda_{ij} \chi_{ij} + \sum_i \lambda_i \chi_i$$

$$= \sum_{i<j} \lambda_{ij} \chi_i \chi_j + \sum_i \lambda_i \chi_i.$$

Here I have used

$$\chi_{ij} = \chi_i \chi_j$$

which follows from

$$\alpha_{ij} = \alpha_i \ \& \ \alpha_j.$$

Now using the formula for χ given above,

$$\chi_i = \tfrac{1}{2}(r_i + 1) = \begin{cases} 1 & \text{if } r_i = +, \\ 0 & \text{if } r_i = -. \end{cases}$$

If we regard the variables of the system to be the χ_i instead of the r_i, this formula for U can be identified with minus the formula for energy, E, in the Boltzmann machine formalism (see Chapter 7). The mapping takes the harmony feature r_i to the Boltzmann node χ_i, the harmony parameter λ_{ij} to the Boltzmann weight w_{ij}, and minus the parameter λ_i to the threshold θ_i. Harmony theory's estimated probability for states of the environment, e^U, is then mapped onto the Boltzmann machine's estimate, e^{-E}. For the isomorphism to be complete, the value of λ that arises from learning in harmony theory must map onto the weights and thresholds given by the Boltzmann machine learning procedure. This is established by the following theorem, which also incorporates the preceding results.

Theorem 4. Consider a cognitive system with the above set of first- and second-order observables, O. Then the weights $\{w_{ij}\}_{i<j}$ and thresholds $\{\theta_i\}_i$ learned by the Boltzmann machine are related to the parameters λ generated by the trace learning procedure by the relations $w_{ij} = \lambda_{ij}$ and $\theta_i = -\lambda_i$. It follows that the Boltzmann machine energy function, E, is equal to $-U$, and the Boltzmann machine's estimated probabilities for environmental states are the same as those of the cognitive system.

This result shows that the Boltzmann criterion of minimizing the information-theoretic distance, G, between the environmental and estimated distributions, subject to the constraint that the estimated distribution be a Gibbs distribution determined by a quadratic function, $-E$, is a consequence of the harmony theory criterion of minimizing the information of the estimated distribution subject to environmental constraints, in the special case that these constraints are no higher than second order.

Proofs of the Theorems

Theorem 1. Part A: The desired maximum-entropy distribution π is the one that maximizes $S(\pi)$ subject to the constraints

$$\sum_{\mathbf{r} \in R} \pi(\mathbf{r}) = 1$$

and

$$<\chi_\alpha>_\pi = p_\alpha$$

where $< >_\pi$ denotes the expected value with respect to the distribution π, and $\{p_\alpha\}_{\alpha \in O}$ are the observable statistics of the environment.

We introduce the Lagrange multipliers λ and λ_α (see, for example, Thomas, 1968) and solve for the values of $\pi(\mathbf{r})$ obeying

$$0 = \frac{\partial}{\partial \pi(\mathbf{r})}\left\{ \sum_{\mathbf{r}' \in R} \pi(\mathbf{r}') \ln \pi(\mathbf{r}') - \sum_{\alpha \in O} \lambda_\alpha \left[\sum_{\mathbf{r}' \in R} \chi_\alpha(\mathbf{r}') \pi(\mathbf{r}') - p_\alpha \right] - \lambda \left[\sum_{\mathbf{r}' \in R} \pi(\mathbf{r}') - 1 \right] \right\}.$$

This leads directly to A. Part B: Since χ_α can be expressed as the product of $|\mathbf{k}_\alpha|$ terms each linear in the feature variables, the function U is a polynomial in the features of degree $|\mathbf{k}_\alpha|$. By introducing new variables a_α, U will now be replaced by a quadratic function H. The trick is to write

$$\chi_\alpha(\mathbf{r}) = \begin{cases} 1 & \text{if } \mathbf{r}\cdot\mathbf{k}_\alpha/|\mathbf{k}_\alpha| = 1 \\ 0 & \text{otherwise} \end{cases}$$

as

$$\chi_\alpha(\mathbf{r}) = \max_{a_\alpha \in \{0,1\}} \left[\frac{a_\alpha}{1-\kappa} \left(\mathbf{r}\cdot\mathbf{k}_\alpha/|\mathbf{k}_\alpha| - \kappa \right) \right] = \max_{a_\alpha \in \{0,1\}} \frac{a_\alpha}{1-\kappa} h(\mathbf{r}, \mathbf{k}_\alpha)$$

where κ is chosen close enough to 1 that $\mathbf{r}\cdot\mathbf{k}_\alpha/|\mathbf{k}_\alpha|$ can only exceed κ by equaling 1. This is assured by the condition on κ of the theorem. Now U can be written

$$U(\mathbf{r}) = \sum_{\alpha \in O} \sigma_\alpha \max_{a_\alpha \in \{0,1\}} [a_\alpha h(\mathbf{r}, \mathbf{k}_\alpha)] = \max_{\mathbf{a} \in A} H(\mathbf{a}, \mathbf{r})$$

where the strengths σ_α are simply the Lagrange multipliers, rescaled:

$$\sigma_\alpha = \frac{\lambda_\alpha}{1-\kappa}.$$

Computing the maximum-likelihood completion function c_π requires maximizing $\pi(\mathbf{r}) \propto e^{U(\mathbf{r})}$ over those $\mathbf{r} \in R$ that are completions of the input ι. This is equivalent to maximizing $U(\mathbf{r})$, since the exponential function is monotonically increasing. But,

$$\max_{\mathbf{r} \supset \iota} U(\mathbf{r}) = \max_{\mathbf{r} \supset \iota} \max_{\mathbf{a} \in A} H(\mathbf{r}, \mathbf{a}).$$

Thus the maximum-likelihood completion function $c_\pi = c_{p_U}$ determined by π, the Gibbs distribution determined by U, is the same as the maximum-likelihood completion function c_{p_H} determined by p_H, the Gibbs distribution determined by H. Note that p_H is a distribution

on the enlarged space $M = R \times A$. For Theorem 3, the conditions determining the Lagrange multipliers (strengths) will be examined.

Theorem 2. Part A: This classic result has, since Metropolis et al. (1953), provided the foundation for the computer simulation of thermal systems. We will prove that the stochastic process determined by any probability distribution p always converges to p. The stochastic process $\boldsymbol{x}$ is a *Markov process with a stationary transition probability matrix.* (The probability of making a transition from one state to another is time-independent. This is not true of a process in which variables are updated in a fixed sequence rather than by randomly selecting a variable according to some fixed probability distribution. For the sequential updating process, Theorem 2A still holds, but the proof is less direct [see, for example, Smolensky, 1981]). Since only one variable can change per time step, $|X|$ steps are required to completely change from one state to another. However in $|X|$ time steps, any state has a nonzero probability of changing to any other state. In the language of stochastic processes, this means that the process is *irreducible.* It is an important result from the theory of stochastic processes that in a finite state space any irreducible Markov process approaches, in the above sense, a unique limiting distribution as $t \to \infty$ (Lamperti, 1977). It remains only to show that this limiting distribution is p. The argument now is that p is a *stationary distribution* of the process. This means that if at any time t the distribution of states of the process is p, then at the next time $t+1$ (and hence at all later times) the distribution will remain p. Once p is known to be stationary, it follows that p is the unique limiting distribution, since we could always start the process with distribution p, and it would have to converge to the limiting distribution, all the while remaining in the stationary distribution p. To show that p is a stationary distribution for the process, we assume that at time t the distribution of states is p. The distribution at time $t+1$ is then

$$\begin{aligned}\mathrm{pr}(\boldsymbol{x}(t+1) = \mathbf{x}) &= \sum_{\mathbf{x}' \in X_{\mathbf{x}}} \mathrm{pr}(\boldsymbol{x}(t) = \mathbf{x}')\ \mathrm{pr}(\boldsymbol{x}(t+1) = \mathbf{x} \mid \boldsymbol{x}(t) = \mathbf{x}') \\ &= \sum_{\mathbf{x}' \in X_{\mathbf{x}}} p(\mathbf{x}')\ W_{\mathbf{x}'\mathbf{x}}.\end{aligned}$$

The sum here runs over $X_{\mathbf{x}}$, the set of states that differ from $\mathbf{x}$ in at most one coordinate; for the remaining states, the one time-step transition probability $W_{\mathbf{x}'\mathbf{x}} = \mathrm{pr}(\boldsymbol{x}(t+1) = \mathbf{x} | \boldsymbol{x}(t) = \mathbf{x}')$ is zero. Next we use the important *detailed balance condition*,

$$p(\mathbf{x}')\ W_{\mathbf{x}'\mathbf{x}} = p(\mathbf{x})\ W_{\mathbf{x}\mathbf{x}'}$$

which states that in an ensemble of systems with states distributed according to p, the number of transitions from $\mathbf{x}'$ to $\mathbf{x}$ is equal to the number from $\mathbf{x}$ to $\mathbf{x}'$. Detailed balance holds because, for the nontrivial case in which $\mathbf{x}'$ and $\mathbf{x}$ differ in the single coordinate v, the transition matrix W determined by the distribution p is

$$W_{\mathbf{x}'\mathbf{x}} = P_v \frac{p(\mathbf{x})}{p(\mathbf{x}) + p(\mathbf{x}')}$$

where P_v is the probability of selecting for update the coordinate v. Now we have

$$\begin{aligned} \mathrm{pr}(x(t+1) = \mathbf{x}) &= \sum_{\mathbf{x}' \in X_{\mathbf{x}}} p(\mathbf{x}')\, W_{\mathbf{x}'\mathbf{x}} = \sum_{\mathbf{x}' \in X_{\mathbf{x}}} p(\mathbf{x})\, W_{\mathbf{x}\mathbf{x}'} \\ &= p(\mathbf{x}) \sum_{\mathbf{x}' \in X_{\mathbf{x}}} W_{\mathbf{x}\mathbf{x}'} = p(\mathbf{x}). \end{aligned}$$

The last equality follows from

$$\sum_{\mathbf{x}' \in X_{\mathbf{x}}} W_{\mathbf{x}\mathbf{x}'} = 1$$

which simply states that the probability of a transition from $\mathbf{x}$ to *some* state $\mathbf{x}'$ is 1. The conclusion is that the probability distribution at time $t+1$ remains p, which is therefore a stationary distribution.

Part B: Part A assures us that with infinite patience we can arbitrarily well approximate the distribution p_T at any finite temperature T. It seems intuitively clear that with still further patience we could sequentially approximate in one long stochastic process a series of distributions p_{T_t} with temperatures T_t monotonically decreasing to zero. This process would presumably converge to the zero-temperature distribution that corresponds to the maximum-likelihood completion function. A proof that this is true, provided

$$T_t > C/\ln t$$

for suitable C, can be found in S. Geman and D. Geman (1984).

Theorem 3. We now pick up the analysis from the end of the proof of Theorem 1.

Lemma. (S. Geman, personal communication, 1984.) The values of the Lagrange multipliers $\boldsymbol{\lambda} = \{\lambda_\alpha\}_{\alpha \in O}$ defining the function U of Theorem 2 are those that minimize the convex function:

$$F(\lambda) = \ln Z_V(\lambda) = \ln \sum_{r \in R} e^{\sum_{\alpha \in O} \lambda_\alpha [\chi_\alpha(r) - p_\alpha]}.$$

Proof of Lemma: Note that

$$p_U(r) = p_V(r) = Z_V(\lambda)^{-1} e^{V(r)}$$

where

$$V(r) = \sum_{\alpha \in O} \lambda_\alpha [\chi_\alpha(r) - p_\alpha] = U(r) - \sum_{\alpha \in O} \lambda_\alpha p_\alpha.$$

From this it follows that the gradient of F is

$$\frac{\partial F}{\partial \lambda_\alpha} = < \chi_\alpha >_{p_U} - p_\alpha$$

The constraint that λ enforces is precisely that this vanish for all α; then $p_U = \pi$. Thus the correct λ is a critical point of F. To see that in fact the correct λ is a minimum of F, we show that F has a positive-definite matrix of second-partial derivatives and is therefore convex. It is straightforward to verify that the quadratic form

$$\sum_{\alpha,\alpha' \in O} q_\alpha \frac{\partial^2 F}{\partial \lambda_\alpha \partial \lambda_{\alpha'}} q_{\alpha'}$$

is the variance

$$< (Q - <Q>_{p_U})^2 >_{p_U}$$

of the random variable Q defined by $Q(r) = \sum_{\alpha \in O} q_\alpha \chi_\alpha(r)$. This variance is clearly nonnegative definite. That Q cannot vanish is assured by the assumption that the χ_α are linearly independent. Since a Gibbs distribution p_U is nowhere zero, this means that the variance of Q is positive, so the Lemma is proved.

Proof of Theorem 3: Since F is convex, we can find its minimum, λ, by gradient descent from any starting point. The process of learning the correct λ, then, can proceed in time according to the gradient descent equation

$$\frac{d\lambda_\alpha}{dt} \propto -\frac{\partial F}{\partial \lambda_\alpha} = -(<\chi_\alpha>_{p_U} - p_\alpha) = <\chi_\alpha>_p - <\chi_\alpha>_{p_U}$$

where it is understood that the function U changes as λ changes. The two phases of the trace learning procedure generate the two terms in this equation. In the environmental observation phase, the increment

$<\chi_\alpha>_p$ is estimated; in the environmental simulation phase, the decrement $<\chi_\alpha>_{p_U}$ is estimated (following Theorem 2). By hypothesis, these estimates are accurate. (That is, this theorem treats the ideal case of perfect samples, with sample means equal to the true population means.) Thus λ will converge to the correct value. The proportional relation between σ and λ was derived in the proof of Theorem 1.

Theorem 4. The proof of Theorem 3 shows that the trace learning procedure does gradient descent in the function F. The Boltzmann learning procedure does gradient descent in the function G:

$$G(\lambda) = -\sum_{\mathbf{r}} p(\mathbf{r}) \ln \frac{p_U(\mathbf{r})}{p(\mathbf{r})}$$

where, as always, the function U implicitly depends on λ. Theorem 4 will be proved by showing that in fact F and G differ by a constant independent of λ, and therefore they define the same gradient descent trajectories. From the above definition of V, we have

$$V(\mathbf{r}) = U(\mathbf{r}) - \sum_{\alpha \in O} \lambda_\alpha <\chi_\alpha> = U(\mathbf{r}) - <U>$$

where, here and henceforth, $< >$ denotes expectation values with respect to the environmental distribution p. This implies

$$\sum_{\mathbf{r}} e^{V(\mathbf{r})} = e^{-<U>} \sum_{\mathbf{r}} e^{U(\mathbf{r})},$$

i.e.,

$$Z_V = Z_U \, e^{-<U>}.$$

By the definition of F,

$$F = \ln Z_V = \ln Z_U - <U> = < \ln Z_U - U >.$$

To evaluate the last quantity in angle brackets, note that

$$p_U(\mathbf{r}) = Z_U^{-1} \, e^{U(\mathbf{r})}$$

implies

$$\ln p_U(\mathbf{r}) = -\ln Z_U + U(\mathbf{r})$$

so that the preceding equation for F becomes

$$F = < \ln Z_U - U > = - < \ln p_U > = - \sum_{\mathbf{r}} p(\mathbf{r}) \ln p_U(\mathbf{r}).$$

Now,

$$G = -\sum_{\mathbf{r}} p(\mathbf{r}) \ln p_U(\mathbf{r}) + \sum_{\mathbf{r}} p(\mathbf{r}) \ln p(\mathbf{r}),$$

so we have

$$G(\lambda) = F(\lambda) - S(p).$$

Thus, as claimed, G is just F minus a constant that is independent of λ: the entropy of the environment.