COMPUTER VISION, GRAPHICS, AND IMAGE PROCESSING 37, 54–115 (1987)

A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine

GAIL A. CARPENTER*

Department of Mathematics, Northeastern University, Boston, Massachusetts 02215 and Center for Adaptive Systems, Department of Mathematics, Boston University, Boston, Massachusetts 02215

AND

STEPHEN GROSSBERG†

Center for Adaptive Systems, Department of Mathematics, Boston University, Boston, Massachusetts 02215

Received March 3, 1986

A neural network architecture for the learning of recognition categories is derived. Real-time network dynamics are completely characterized through mathematical analysis and computer simulations. The architecture self-organizes and self-stabilizes its recognition codes in response to arbitrary orderings of arbitrarily many and arbitrarily complex binary input patterns. Top–down attentional and matching mechanisms are critical in self-stabilizing the code learning process. The architecture embodies a parallel search scheme which updates itself adaptively as the learning process unfolds. After learning self-stabilizes, the search process is automatically disengaged. Thereafter input patterns directly access their recognition codes without any search. Thus recognition time does not grow as a function of code complexity. A novel input pattern can directly access a category if it shares invariant properties with the set of familiar exemplars of that category. These invariant properties emerge in the form of learned critical feature patterns, or prototypes. The architecture possesses a context-sensitive self-scaling property which enables its emergent critical feature patterns to form. They detect and remember statistically predictive configurations of featural elements which are derived from the set of all input patterns that are ever experienced. Four types of attentional process—priming, gain control, vigilance, and intermodal competition—are mechanistically characterized. Top—down priming and gain control are needed for code matching and self-stabilization. Attentional vigilance determines how fine the learned categories will be. If vigilance increases due to an environmental disconfirmation, then the system automatically searches for and learns finer recognition categories. A new nonlinear matching law (the $\frac{2}{3}$ Rule) and new nonlinear associative laws (the Weber Law Rule, the Associative Decay Rule, and the Template Learning Rule) are needed to achieve these properties. All the rules describe emergent properties of parallel network interactions. The architecture circumvents the noise, saturation, capacity, orthogonality, and linear predictability constraints that limit the codes which can be stably learned by alternative recognition models.

1. INTRODUCTION: SELF-ORGANIZATION OF NEURAL RECOGNITION CODES

A fundamental problem of perception and cognition concerns the characterization of how humans discover, learn, and recognize invariant properties of the environments to which they are exposed. When such recognition codes sponta-

*Supported in part by the Air Force Office of Scientific Research Grants AFOSR 85-0149 and AFOSR 86-F49620-86-C-0037, the Army Research Office Grant ARO DAAG-29-85-K-0095, and the National Science Foundation Grant NSF DMS-84-13119.

†Supported in part by the Air Force Office of Scientific Research Grants AFOSR 85-0149 and AFOSR 86-F49620-86-C-0037 and the Army Research Office Grant ARO DAAG-29-85-K0095.

0734-189X/87 $3.00

neously emerge through an individual's interaction with an environment, the processes are said to undergo *self-organization* [1]. This article develops a theory of how recognition codes are self-organized by a class of neural networks whose qualitative features have been used to analyse data about speech perception, word recognition and recall, visual perception, olfactory coding, evoked potentials, thalamocortical interactions, attentional modulation of critical period termination, and amnesias [2–13]. These networks comprise the *adaptive resonance theory* (ART) which was introduced in Grossberg [8].

This article describes a system of differential equations which completely characterizes one class of ART networks. The network model is capable of self-organizing, self-stabilizing, and self-scaling its recognition codes in response to arbitrary temporal sequences of arbitrarily many input patterns of variable complexity. These formal properties, which are mathematically proven herein, provide a secure foundation for designing a real-time hardware implementation of this class of massively parallel ART circuits.

Before proceeding to a description of this class of ART systems, we summarize some of their major properties and some scientific problems for which they provide a solution.

A. *Plasticity*

Each system generates recognition codes adaptively in response to a series of environmental inputs. As learning proceeds, interactions between the inputs and the system generate new steady states and basins of attraction. These steady states are formed as the system discovers and learns *critical feature patterns*, or prototypes, that represent invariants of the set of all experienced input patterns.

B. *Stability*

The learned codes are dynamically buffered against relentless recoding by irrelevant inputs. The formation of steady states is internally controlled using mechanisms that suppress possible sources of system instability.

C. *Stability–Plasticity Dilemma: Multiple Interacting Memory Systems*

The properties of plasticity and stability are intimately related. An adequate system must be able to adaptively switch between its stable and plastic modes. It must be capable of plasticity in order to learn about significant new events, yet it must also remain stable in response to irrelevant or often repeated events. In order to prevent the relentless degradation of its learned codes by the "blooming, buzzing confusion" of irrelevant experience, an ART system is sensitive to *novelty*. It is capable of distinguishing between familiar and unfamiliar events, as well as between expected and unexpected events.

Multiple interacting memory systems are needed to monitor and adaptively react to the novelty of events. Within ART, interactions between two functionally complementary subsystems are needed to process familiar and unfamiliar events. Familiar events are processed within an attentional subsystem. This subsystem establishes ever more precise internal representations of and responses to familiar events. It also builds up the learned top–down expectations that help to stabilize the learned bottom–up codes of familiar events. By itself, however, the attentional subsystem is unable simultaneously to maintain stable representations of familiar categories and to create new categories for unfamiliar patterns. An isolated attentional subsystem is either rigid and incapable of creating new categories for

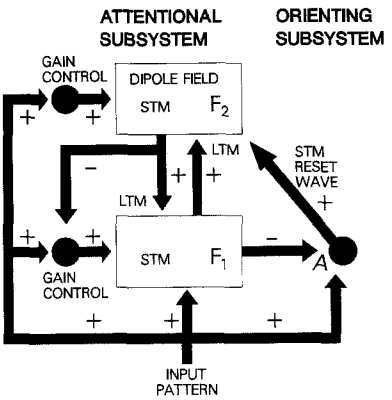

FIG. 1. Anatomy of the attentional-orienting system: Two successive stages, F_1 and F_2, of the attentional subsystem encode patterns of activation in short term memory (STM). Bottom–up and top–down pathways between F_1 and F_2 contain adaptive long term memory (LTM) traces which multiply the signals in these pathways. The remainder of the circuit modulates these STM and LTM processes. Modulation by gain control enables F_1 to distinguish between bottom–up input patterns and top–down priming, or template, patterns, as well as to match these bottom–up and top–down patterns. Gain control signals also enable F_2 to react supraliminally to signals from F_1 while an input pattern is on. The orienting subsystem generates a reset wave to F_2 when mismatches between bottom–up and top–down patterns occur at F_1. This reset wave selectively and enduringly inhibits active F_2 cells until the input is shut off. Variations of this architecture are depicted in Fig. 14.

unfamiliar patterns, or unstable and capable of ceaselessly recoding the categories of familiar patterns in response to certain input environments.

The second subsystem is an orienting subsystem that resets the attentional subsystem when an unfamiliar event occurs. The orienting subsystem is essential for expressing whether a novel pattern is familiar and well represented by an existing recognition code, or unfamiliar and in need of a new recognition code. Figure 1 schematizes the architecture that is analysed herein.

D. *Role of Attention in Learning*

Within an ART system, attentional mechanisms play a major role in self-stabilizing the learning of an emergent recognition code. Our mechanistic analysis of the role of attention in learning leads us to distinguish between four types of attentional mechanism: attentional priming, attentional gain control, attentional vigilance, and intermodality competition. These mechanisms are characterized below.

E. *Complexity*

An ART system dynamically reorganizes its recognition codes to preserve its stability–plasticity balance as its internal representations become increasingly complex and differentiated through learning. By contrast, many classical adaptive pattern recognition systems become unstable when they are confronted by complex input environments. The instabilities of a number of these models are identified in Grossberg [7, 11, 14]. Models which become unstable in response to nontrivial input environments are not viable either as brain models or as designs for adaptive machines.

Unlike many alternative models [15–19], the present model can deal with arbitrary combinations of binary input patterns. In particular, it places no orthogonality

or linear predictability constraints upon its input patterns. The model computations remain sensitive no matter how many input patterns are processed. The model does not require that very small, and thus noise-degradable, increments in memory be made in order to avoid saturation of its cumulative memory. The model can store arbitrarily many recognition categories in response to input patterns that are defined on arbitrarily many input channels. Its memory matrices need not be square, so that no restrictions on memory capacity are imposed by the number of input channels. Finally, all the memory of the system can be devoted to stable recognition learning. It is not the case that the number of stable classifications is bounded by some fraction of the number of input channels or patterns.

Thus a primary goal of the present article is to characterize neural networks capable of self-stabilizing the self-organization of their recognition codes in response to an arbitrarily complex environment of input patterns in a way that parsimoniously reconciles the requirements of plasticity, stability, and complexity.

2. SELF-SCALING COMPUTATIONAL UNITS, SELF-ADJUSTING MEMORY SEARCH, DIRECT ACCESS, AND ATTENTIONAL VIGILANCE

Four properties are basic to the workings of the networks that we characterize herein.

A. *Self-Scaling Computational Units: Critical Feature Patterns*

Properly defining signal and noise in a self-organizing system raises a number of subtle issues. Pattern context must enter the definition so that input features which are treated as irrelevant noise when they are embedded in a given input pattern may be treated as informative signals when they are embedded in a different input pattern. The system's unique learning history must also enter the definition so that portions of an input pattern which are treated as noise when they perturb a system at one stage of its self-organization may be treated as signals when they perturb the same system at a different stage of its self-organization. The present systems automatically self-scale their computational units to embody context- and learning-dependent definitions of signal and noise.

One property of these self-scaling computational units is schematized in Fig. 2. In Fig. 2a, each of the two input patterns is composed of three features. The patterns agree at two of the three features, but disagree at the third feature. A mismatch of one out of three features may be designated as informative by the system. When this occurs, these mismatched features are treated as signals which can elicit learning of distinct recognition codes for the two patterns. Moreover, the mismatched features, being informative, are incorporated into these distinct recognition codes.

In Fig. 2b, each of the two input patterns is composed of 31 features. The patterns are constructed by adding identical subpatterns to the two patterns in Fig. 2a. Thus the input patterns in Fig. 2b disagree at the same features as the input patterns in Fig. 2a. In the patterns of Fig. 2b, however, this mismatch is less important, other things being equal, than in the patterns of Fig. 2a. Consequently, the system may treat the mismatched features as noise. A single recognition code may be learned to represent both of the input patterns in Fig. 2b. The mismatched features would not be learned as part of this recognition code because they are treated as noise.

The assertion that *critical feature patterns* are the computational units of the code learning process summarizes this self-scaling property. The term *critical feature*

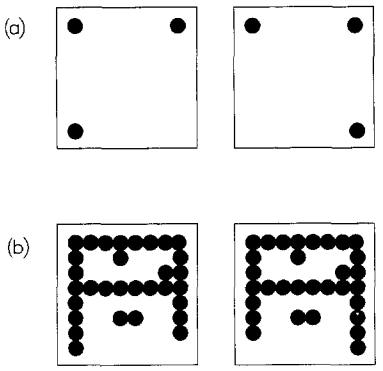

FIG. 2. Self-scaling property discovers critical features in a context-sensitive way: (a) Two input patterns of 3 features mismatch at 1 feature. When this mismatch is sufficient to generate distinct recognition codes for the two patterns, the mismatched features are encoded in LTM as part of the critical feature patterns of these recognition codes. (b) Identical subpatterns are added to the two input patterns in (a). Although the new input patterns mismatch at the same one feature, this mismatch may be treated as noise due to the additional complexity of the two new patterns. Both patterns may thus learn to activate the same recognition code. When this occurs, the mismatched feature is deleted from LTM in the critical feature pattern of the code.

indicates that not all features are treated as signals by the system. The learned units are *patterns* of critical features because the perceptual context in which the features are embedded influences which features will be processed as signals and which features will be processed as noise. Thus a feature may be a critical feature in one pattern (Fig. 2a) and an irrelevant noise element in a different pattern (Fig. 2b).

The need to overcome the limitations of featural processing with some of type of contextually sensitive pattern processing has long been a central concern in the human pattern recognition literature. Experimental studies have led to the general conclusions that "the trace system which underlies the recognition of patterns can be characterized by a central tendency and a boundary" [20, p. 54], and that "just listing features does not go far enough in specifying the knowledge represented in a concept. People also know something about the relations between the features of a concept, and about the variability that is permissible on any feature" [21, p. 83]. We illustrate herein how these properties may be achieved using self-scaling computational units such as critical feature patterns.

B. *Self-Adjusting Memory Search*

No pre-wired search algorithm, such as a search tree, can maintain its efficiency as a knowledge structure evolves due to learning in a unique input environment. A search order that may be optimal in one knowledge domain may become extremely inefficient as that knowledge domain becomes more complex due to learning.

The ART system considered herein is capable of a parallel memory search that adaptively updates its search order to maintain efficiency as its recognition code becomes arbitrarily complex due to learning. This self-adjusting search mechanism is part of the network design whereby the learning process self-stabilizes by engaging the orienting subsystem (Sect. 1C).

None of these mechanisms is akin to the rules of a serial computer program. Instead, the circuit architecture as a whole generates a self-adjusting search order and self-stabilization as emergent properties that arise through system interactions. Once the ART architecture is in place, a little randomness in the initial values of its memory traces, rather than a carefully wired search tree, enables the search to carry on until the recognition code self-stabilizes.

C. *Direct Access to Learned Codes*

A hallmark of human recognition performance is the remarkable rapidity with which familiar objects can be recognized. The existence of many learned recognition codes for alternative experiences does not necessarily interfere with rapid recognition of an unambiguous familiar event. This type of rapid recognition is very difficult to understand using models wherein trees or other serial algorithms need to be searched for longer and longer periods as a learned recognition code becomes larger and larger.

In an ART model, as the learned code becomes globally self-consistent and predictively accurate, the search mechanism is automatically disengaged. Subsequently, no matter how large and complex the learned code may become, familiar input patterns *directly access*, or activate, their learned code, or category. Unfamiliar patterns can also directly access a learned category if they share invariant properties with the critical feature pattern of the category. In this sense, the critical feature pattern acts as a prototype for the entire category. As in human pattern recognition experiments, an input pattern that matches a learned critical feature pattern may be better recognized than any of the input patterns that gave rise to the critical feature pattern [20, 22, 23].

Unfamiliar input patterns which cannot stably access a learned category engage the self-adjusting search process in order to discover a network substrate for a new recognition category. After this new code is learned, the search process is automatically disengaged and direct access ensues.

D. *Environment as a Teacher: Modulation of Attentional Vigilance*

Although an ART system self-organizes its recognition code, the environment can also modulate the learning process and thereby carry out a teaching role. This teaching role allows a system with a fixed set of feature detectors to function successfully in an environment which imposes variable performance demands. Different environments may demand either coarse discriminations or fine discriminations to be made among the same set of objects. As Posner [20, pp. 53–54] has noted:

> If subjects are taught a tight concept, they tend to be very careful about classifying any particular pattern as an instance of that concept. They tend to reject a relatively small distortion of the prototype as an instance, and they rarely classify a pattern as a member of the concept when it is not. On the other hand, subjects learning high-variability concepts often falsely classify patterns as members of the concept, but rarely reject a member of the concept incorrectly... The situation largely determines which type of learning will be superior.

In an ART system, if an erroneous recognition is followed by negative reinforcement, then the system becomes more *vigilant*. This change in vigilance may be interpreted as a change in the system's attentional state which increases its sensitivity to mismatches between bottom–up input patterns and active top–down critical

feature patterns. A vigilance change alters the size of a single parameter in the network. The *interactions* within the network respond to this parameter change by learning recognition codes that make finer distinctions. In other words, if the network erroneously groups together some input patterns, then negative reinforcement can help the network to learn the desired distinction by making the system more vigilant. The system then behaves *as if* has a better set of feature detectors.

The ability of a vigilance change to alter the course of pattern recognition illustrates a theme that is common to a variety of neural processes: a one-dimensional parameter change that modulates a simple nonspecific neural process can have complex specific effects upon high-dimensional neural information processing.

Sections 3–7 outline qualitatively the main operations of the model. Sections 8–11 describe computer simulations which illustrate the model's ability to learn categories. Section 12 defines the model mathematically. The remaining sections characterize the model's properties using mathematical analysis and more computer simulations, with the model hypotheses summarized in Section 18.

3. BOTTOM–UP ADAPTIVE FILTERING AND CONTRAST-ENHANCEMENT IN SHORT TERM MEMORY

We begin by considering the typical network reactions to a single input pattern I within a temporal stream of input patterns. Each input pattern may be the output pattern of a preprocessing stage. Different preprocessing is given, for example, to speech signals and to visual signals before the outcome of such modality-specific preprocessing ever reaches the attentional subsystem. The preprocessed input pattern I is received at the stage F_1 of an attentional subsystem. Pattern I is transformed into a pattern X of activation across the nodes, or abstract "feature detectors," of F_1 (Fig. 3). The transformed pattern X represents a pattern in short term memory (STM). In F_1 each node whose activity is sufficiently large generates

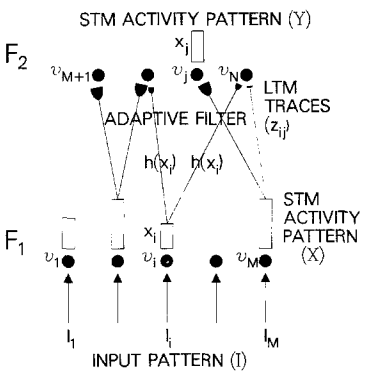

FIG. 3. Stages of bottom–up activation: The input pattern I generates a pattern of STM activation X across F_1. Sufficiently active F_1 nodes emit bottom–up signals to F_2. This signal pattern S is gated by long term memory (LTM) traces within the $F_1 \rightarrow F_2$ pathways. The LTM-gated signals are summed before activating their target nodes in F_2. This LTM-gated and summed signal pattern T generates a pattern of activation Y across F_2. The nodes in F_1 are denoted by $v_1, v_2, \ldots, v_M$. The nodes in F_2 are denoted by $v_{M+1}, v_{M+2}, \ldots, v_N$. The input to node v_i is denoted by I_i. The STM activity of node v_i is denoted by x_i. The LTM trace of the pathway from v_i to v_j is denoted by z_{ij}.

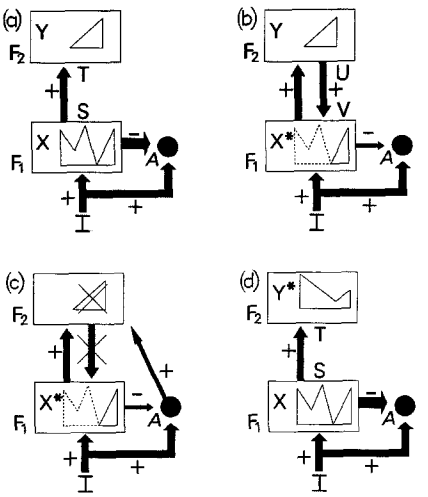

FIG. 4. Search for a correct F_2 code: (a) The input pattern I generates the specific STM activity pattern X at F_1 as it nonspecifically activates A. Pattern X both inhibits A and generates the output signal pattern S. Signal pattern S is transformed into the input pattern T, which activates the STM pattern Y across F_2. (b) Pattern Y generates the top–down signal pattern U which is transformed into the template pattern V. If V mismatches I at F_1, then a new STM activity pattern X^* is generated at F_1. The reduction in total STM activity which occurs when X is transformed into X^* causes a decrease in the total inhibition from F_1 to A. (c) Then the input-driven activation of A can release a nonspecific arousal wave to F_2, which resets the STM pattern Y at F_2. (d) After Y is inhibited, its top–down template is eliminated, and X can be reinstated at F_1. Now X once again generates input pattern T to F_2, but since Y remains inhibited T can activate a different STM pattern Y^* at F_2. If the top–down template due to Y^* also mismatches I at F_1, then the rapid search for an appropriate F_2 code continues.

excitatory signals along pathways to target nodes at the next processing stage F_2. A pattern X of STM activities across F_1 hereby elicits a pattern S of output signals from F_1. When a signal from a node in F_1 is carried along a pathway to F_2, the signal is multiplied, or *gated*, by the pathway's long term memory (LTM) trace. The LTM-gated signal (i.e., signal times LTM trace), not the signal alone, reaches the target node. Each target node sums up of all of its LTM-gated signals. In this way, pattern S generates a pattern T of LTM-gated and summed input signals to F_2 (Fig. 4a). The transformation from S to T is called an *adaptive filter*.

The input pattern T to F_2 is quickly transformed by interactions among the nodes of F_2. These interactions contrast-enhance the input pattern T. The resulting pattern of activation across F_2 is a new pattern Y. The contrast-enhanced pattern Y, rather than the input pattern T, is stored in STM by F_2.

A special case of this contrast-enhancement process is one in which F_2 chooses the node which receives the largest input. The chosen node is the only one that can store activity in STM. In general, the contrast enhancing transformation from T to Y enables more than one node at a time to be active in STM. Such transformations are designed to simultaneously represent in STM several groupings, or chunks, of an input pattern [9, 11, 24–26]. When F_2 is designed to make a choice in STM, it selects that global grouping of the input pattern which is preferred by the adaptive filter. This process automatically enables the network to partition all the input

patterns which are received by F_1 into disjoint sets of recognition categories, each corresponding to a particular node (or "pointer," or "index") in F_2. Such a categorical mechanism is both interesting in itself and a necessary prelude to the analysis of recognition codes in which multiple groupings of X are simultaneously represented by Y. In the example that is characterized in this article, level F_2 is designed to make a choice.

All the LTM traces in the adaptive filter, and thus all learned past experiences of the network, are used to determine the recognition code Y via the transformation $I \rightarrow X \rightarrow S \rightarrow T \rightarrow Y$. However, only those nodes of F_2 which maintain stored activity in the STM pattern Y can elicit new learning at contiguous LTM traces. Because the recognition code Y is a more contrast-enhanced pattern than T, many F_2 nodes which receive positive inputs ($I \rightarrow X \rightarrow S \rightarrow T$) may not store any STM activity ($T \rightarrow Y$). The LTM traces in pathways leading to these nodes thus influence the recognition event but are not altered by the recognition event. Some memories which influence the focus of attention are not themselves attended.

4. TOP–DOWN TEMPLATE MATCHING AND STABILIZATION OF CODE LEARNING

As soon as the bottom–up STM transformation $X \rightarrow Y$ takes place, the STM activities Y in F_2 elicit a top–down excitatory signal pattern U back to F_1 (Fig. 4b). Only sufficiently large STM activities in Y elicit signals in U along the feedback pathways $F_2 \rightarrow F_1$. As in the bottom–up adaptive filter, the top–down signals U are also gated by LTM traces and the LTM-gated signals are summed at F_1 nodes. The pattern U of output signals from F_2 hereby generates a pattern V of LTM-gated and summed input signals to F_1. The transformation from U to V is thus also an adaptive filter. The pattern V is called a *top–down template*, or *learned expectation*.

Two sources of input now perturb F_1: the bottom–up input pattern I which gave rise to the original activity pattern X, and the top–down template pattern V that resulted from activating X. The activity pattern X^* across F_1 that is induced by I and V taken together is typically different from the activity pattern X that was previously induced by I alone. In particular, F_1 acts to match V against I. The result of this matching process determines the future course of learning and recognition by the network.

The entire activation sequence

$$I \rightarrow X \rightarrow S \rightarrow T \rightarrow Y \rightarrow U \rightarrow V \rightarrow X^* \tag{1}$$

takes place very quickly relative to the rate with which the LTM traces in either the bottom–up adaptive filter $S \rightarrow T$ or the top–down adaptive filter $U \rightarrow V$ can change. Even though none of the LTM traces changes during such a short time, their prior learning strongly influences the STM patterns Y and X^* that evolve within the network by determining the transformations $S \rightarrow T$ and $U \rightarrow V$. We now discuss how a match or mismatch of I and V at F_1 regulates the course of learning in response to the pattern I, and in particular solves the stability-plasticity dilemma (Sect. 1C).

5. INTERACTIONS BETWEEN ATTENTIONAL AND ORIENTING SUBSYSTEMS: STM RESET AND SEARCH

In Fig. 4a, an input pattern I generates an STM activity pattern X across F_1. The input pattern I also excites the orienting subsystem A, but pattern X at F_1 inhibits A before it can generate an output signal. Activity pattern X also elicits an output pattern S which, via the bottom–up adaptive filter, instates an STM activity pattern Y across F_2. In Fig. 4b, pattern Y reads a top–down template pattern V into F_1. Template V mismatches input I, thereby significantly inhibiting STM activity across F_1. The amount by which activity in X is attenuated to generate X^* depends upon how much of the input pattern I is encoded within the template pattern V.

When a mismatch attenuates STM activity across F_1, the total size of the inhibitory signal from F_1 to A is also attenuated. If the attenuation is sufficiently great, inhibition from F_1 to A can no longer prevent the arousal source A from firing. Fig. 4c depicts how disinhibition of A releases an arousal burst to F_2 which equally, or nonspecifically, excites all the F_2 cells. The cell populations of F_2 react to such an arousal signal in a state-dependent fashion. In the special case that F_2 chooses a single population for STM storage, the arousal burst selectively inhibits, or resets, the active population in F_2. This inhibition is long-lasting. One physiological design for F_2 processing which has these properties is a *gated dipole field* [10, 27]. A gated dipole field consists of opponent processing channels which are gated by habituating chemical transmitters. A nonspecific arousal burst induces selective and enduring inhibition of active populations within a gated dipole field.

In Fig. 4c, inhibition of Y leads to removal of the top–down template V, and thereby terminates the mismatch between I and V. Input pattern I can thus reinstate the original activity pattern X across F_1, which again generates the output pattern S from F_1 and the input pattern T to F_2. Due to the enduring inhibition at F_2, the input pattern T can no longer activate the original pattern Y at F_2. A new pattern Y^* is thus generated at F_2 by I (Fig. 4d). Despite the fact that some F_2 nodes may remain inhibited by the STM reset property, the new pattern Y^* may encode large STM activities. This is because level F_2 is designed so that its total suprathreshold activity remains approximately constant, or normalized, despite the fact that some of its nodes may remain inhibited by the STM reset mechanism. This property is related to the limited capacity of STM. A physiological process capable of achieving the STM normalization property is based upon on-center off-surround feedback interactions among cells obeying membrane equations [10, 28].

The new activity pattern Y^* reads out a new top–down template pattern V^*. If a mismatch again occurs at F_1, the orienting subsystem is again engaged, thereby leading to another arousal-mediated reset of STM at F_2. In this way, a rapid series of STM matching and reset events may occur. Such an STM matching and reset series controls the system's search of LTM by sequentially engaging the novelty-sensitive orienting subsystem. Although STM is reset sequentially in time via this mismatch-mediated, self-terminating LTM search process, the mechanisms which control the LTM search are all parallel network interactions, rather than serial algorithms. Such a parallel search scheme continuously adjusts itself to the system's evolving LTM codes. In general, the spatial configuration of LTM codes depends upon both the system's initial configuration and its unique learning history, and hence cannot be predicted a priori by a pre-wired search algorithm. Instead, the mismatch-mediated engagement of the orienting subsystem realizes the type of self-adjusting search that was described in Section 2B.

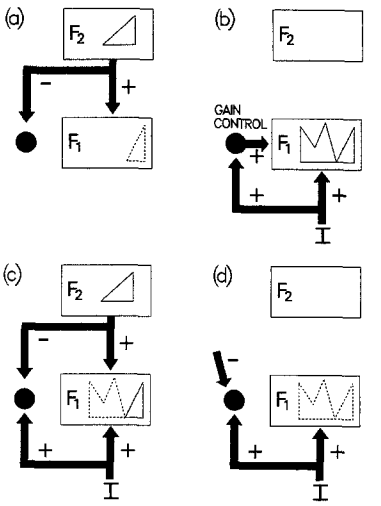

FIG. 5. Matching by the $\frac{2}{3}$ Rule: (a) A top–down template from F_2 inhibits the attentional gain control source as it subliminally primes target F_1 cells. (b) Only F_1 cells that receive bottom–up inputs and gain control signals can become supraliminally active. (c) When a bottom–up input pattern and a top–down template are simultaneously active, only those F_1 cells that receive inputs from both sources can become supraliminally active. (d) Intermodality inhibition can shut off the F_1 gain control source and thereby prevent a bottom–up input from supraliminally activating F_1. Similarly, disinhibition of the F_1 gain control source may cause a top–down prime to become supraliminal.

The mismatched-mediated search of LTM ends when an STM pattern across F_2 reads out a top–down template which matches I, to the degree of accuracy required by the level of attentional vigilance (Sect. 2D), or which has not yet undergone any prior learning. In the latter case, a new recognition category is then established as a bottom–up code and top–down template are learned.

6. ATTENTIONAL GAIN CONTROL AND ATTENTIONAL PRIMING

Further properties of the top–down template matching process can be derived by considering its role in the regulation of attentional priming. Consider, for example, a situation in which F_2 is activated by a level other than F_1 before F_1 can be activated by a bottom–up input (Fig. 5a). In such a situation, F_2 can generate a top-down template V to F_1. The level F_1 is then primed, or sensitized, to receive a bottom–up input that may or may not match the active expectancy. As depicted in Fig. 5a, level F_1 can be primed to receive a bottom–up input without necessarily eliciting suprathreshold output signals in response to the priming expectancy.

On the other hand, an input pattern I must be able to generate a suprathreshold activity pattern X even if no top–down expectancy is active across F_1 (Figs. 4a and 5b). How does F_1 know that it should generate a suprathreshold reaction to a bottom–up input pattern but not to a top–down input pattern? In both cases, excitatory input signals stimulate F_1 cells. Some auxiliary mechanism must exist to distinguish between bottom–up and top–down inputs. This auxiliary mechanism is called *attentional gain control* to distinguish it from *attentional priming* by the top–down template itself (Fig. 5a). While F_2 is active, the attentional priming mechanism delivers *excitatory specific learned* template patterns to F_1. The atten-

tional gain control mechanism has an *inhibitory nonspecific unlearned* effect on the sensitivity with which F_1 responds to the template pattern, as well as to other patterns received by F_1. The attentional gain control process enables F_1 to tell the difference between bottom–up and top–down signals.

7. MATCHING: THE $\frac{2}{3}$ RULE

A rule for pattern matching at F_1, called the $\frac{2}{3}$ Rule, follows naturally from the distinction between attentional gain control and attentional priming. It says that two out of three signal sources must activate an F_1 node in order for that node to generate suprathreshold output signals. In Fig. 5a, during top–down processing, or priming, the nodes of F_1 receive inputs from at most one of their three possible input sources. Hence no cells in F_1 are supraliminally activated by the top–down template. In Fig. 5b, during bottom–up processing, a suprathreshold node in F_1 is one which receives both a specific input from the input pattern I and a nonspecific excitatory signal from the gain control channel. In Fig. 5c, during the matching of simultaneous bottom–up and top–down patterns, the nonspecific gain control signal to F_1 is inhibited by the top–down channel. Nodes of F_1 which receive sufficiently large inputs from both the bottom–up and the top–down signal patterns generate suprathreshold activities. Nodes which receive a bottom–up input or a top–down input, but not both, cannot become suprathreshold: mismatched inputs cannot generate suprathreshold activities. Attentional gain control thus leads to a matching process whereby the addition of top–down excitatory inputs to F_1 can lead to an overall decrease in F_1's STM activity (Figs. 4a and b). Figure 5d shows how competitive interactions across modalities can prevent F_1 from generating a supraliminal reaction to bottom–up signals when attention shifts from one modality to another.

8. CODE INSTABILITY AND CODE STABILITY

The importance of using the $\frac{2}{3}$ Rule for matching is now illustrated by describing how its absence can lead to a temporally unstable code (Fig. 6a). The system becomes unstable when the inhibitory top–down attentional gain control signals (Fig. 5c) are too small for the $\frac{2}{3}$ Rule to hold at F_1. Larger attentional gain control signals restore code stability by reinstating the $\frac{2}{3}$ Rule (Fig. 6b). Figure 6b also illustrates how a novel exemplar can directly access a previously established category; how the category in which a given exemplar is coded can be influenced by the categories which form to encode very different exemplars; and how the network responds to exemplars as coherent groupings of features, rather than to isolated feature matches or mismatches.

Code Instability Example

In Fig. 6, four input patterns, A, B, C, and D, are periodically presented in the order $ABCAD$. Patterns B, C, and D are all subsets of A. The relationships among the inputs that make the simulation work are as follows: $D \subset C \subset A$; $B \subset A$; $B \cap C = \phi$; and $|D| < |B| < |C|$, where $|I|$ denotes the number of features in input pattern I. The choice of input patterns in Fig. 6 is thus one of infinitely many examples in which, without the $\frac{2}{3}$ Rule, an alphabet of four input patterns cannot be stably coded.

The numbers $1, 2, 3, \ldots$, listed at the left in Fig. 6 itemize the presentation order. The next column, labeled BU for Bottom–Up, describes the input pattern that was

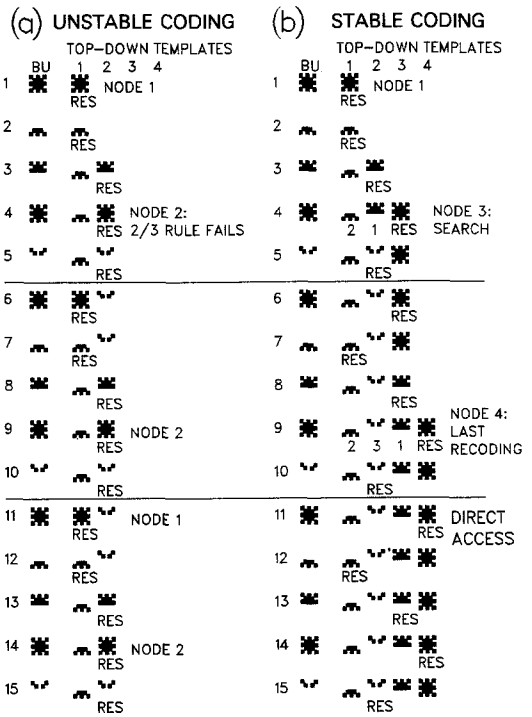

FIG. 6. Stabilization of categorical learning by the $\frac{2}{3}$ Rule: In both (a) and (b), four input patterns A, B, C, and D are presented repeatedly in the list order $ABCAD$. In (a), the $\frac{2}{3}$ Rule is violated because the top–down inhibitory gain control mechanism is weak (Fig. 5c). Pattern A is periodically coded by v_{M+1} and v_{M+2}. It is never coded by a single stable category. In (b), the $\frac{2}{3}$ Rule is restored by strengthening the top–down inhibitory gain control mechanism. After some initial recoding during the first two presentations of $ABCAD$, all patterns directly access distinct stable categories. A black square in a template pattern designates that the corresponding top–down LTM trace is large. A blank square designates that the LTM trace is small.

presented on each trial. Each Top–Down Template column corresponds to a different node in F_2. If M nodes $v_1, v_2, \ldots, v_M$ exist in F_1, then the F_2 nodes are denoted by $v_{M+1}, v_{M+2}, \ldots, v_N$. Column 1 corresponds to node v_{M+1}, column 2 corresponds to node v_{M+2}, and so on. Each row summarizes the network response to its input pattern. The symbol RES, which stands for *resonance*, designates the node in F_2 which codes the input pattern on that trial. For example, v_{M+2} codes pattern C on trial 3, and v_{M+1} codes pattern B on trial 7. The patterns in a given row describe the templates after learning has equilibrated on that trial.

In Fig. 6a, input pattern A is periodically recoded. On trial 1, it is coded by v_{M+1}; on trial 4, it is coded by v_{M+2}; on trial 6, it is coded by v_{M+1}; on trial 9, it is coded by v_{M+2}. This alternation in the nodes v_{M+1} and v_{M+2} which code pattern A repeats indefinitely.

Violation of the $\frac{2}{3}$ Rule occurs on trials 4, 6, 8, 9, and so on. This violation is illustrated by comparing the template of v_{M+2} on trials 3 and 4. On trial 3, the template of v_{M+2} is coded by pattern C, which is a subset of pattern A. On trial 4, pattern A is presented and directly activates node v_{M+2}. Since the inhibitory

top–down gain control is too weak to quench the mismatched portion of the input, pattern A remains supraliminal in F_1 even after the template C is read out from v_{M+2}. No search is elicited by the mismatch of pattern A and its subset template C. Consequently the template of v_{M+2} is recoded from pattern C to its superset pattern A.

Code Stability Example

In Fig. 6b, the $\frac{2}{3}$ Rule does hold because the inhibitory top–down attentional gain control channel is strengthened. Thus the network experiences a sequence of recodings that ultimately stabilizes. In particular, on trial 4, node v_{M+2} reads-out the template C, which mismatches the input pattern A. Here, a search is initiated, as indicated by the numbers beneath the template symbols in row 4. First, v_{M+2}'s template C mismatches A. Then v_{M+1}'s template B mismatches A. Finally A activates the uncommitted node v_{M+3}, which resonates with F_1 as it learns the template A.

In Fig. 6b, pattern A is coded by v_{M+1} on trial 1; by v_{M+3} on trials 4 and 6; and by v_{M+4} on trial 9. Note that the self-adjusting search order in response to A is different on trials 4 and 9 (Sect. 2B). On all future trials, input pattern A is coded by v_{M+4}. Moreover, all the input patterns A, B, C, and D have learned a stable code by trial 9. Thus the code self-stabilizes by the second run through the input list *ABCAD*. On trials 11–15, and on all future trials, each input pattern chooses a different code ($A \rightarrow v_{M+4}$; $B \rightarrow v_{M+1}$; $C \rightarrow v_{M+3}$; $D \rightarrow v_{M+2}$). Each pattern belongs to a separate category because the vigilance parameter (Sect. 2D) was chosen to be large in this example. Moreover, after code learning stabilizes, each input pattern directly activates its node in F_2 without undergoing any additional search (Sect. 2C). Thus after trial 9, only the "RES" symbol appears under the top–down templates. The patterns shown in any row between 9 and 15 provide a complete description of the learned code.

Examples of how a novel exemplar can activate a previously learned category are found on trials 2 and 5 in Figs. 6a and b. On trial 2 pattern B is presented for the first time and directly accesses the category coded by v_{M+1}, which was previously learned by pattern A on trial 1. In other words, B activates the same categorical "pointer," or "marker," or "index" as A. In so doing, B may change the categorical template, which determines which input patterns will also be coded by this index on future trials. The category does not change, but its invariants may change.

9. USING CONTEXT TO DISTINGUISH SIGNAL FROM NOISE IN PATTERNS OF VARIABLE COMPLEXITY

The simulation in Fig. 7 illustrates how, at a fixed vigilance level, the network automatically rescales its matching criterion in response to inputs of variable complexity (Sect. 2A). On the first four trials, the patterns are presented in the order *ABAB*. By trial 2, coding is complete. Pattern A directly accesses node v_{M+1} on trial 3, and pattern B directly accesses node v_{M+2} on trial 4. Thus patterns A and B are coded by different categories. On trials 5–8, patterns C and D are presented in the order *CDCD*. Patterns C and D are constructed from patterns A and B, respectively, by adding identical upper halves to A and B. Thus, pattern C differs from pattern D at the same locations where pattern A differs from pattern B. Due to the addition of these upper halves, the network does not code C in the category v_{M+1} of

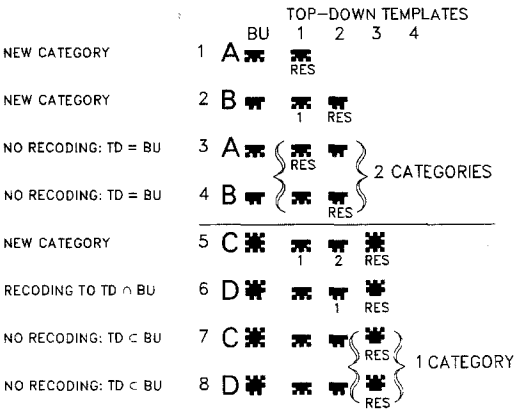

FIG. 7. Distinguishing noise from patterns for inputs of variable complexity: Input patterns A and B are coded by the distinct category nodes v_{M+1} and v_{M+2}, respectively. Input patterns C and D include A and B as subsets, but also possess identical subpatterns of additional features. Due to this additional pattern complexity, C and D are coded by the same category node v_{M+3}. At this vigilance level ($\rho = 0.8$), the network treats the difference between C and D as noise, and suppresses the discordant elements in the v_{M+3} template. By contrast, it treats the difference between A and B as informative, and codes the difference in the v_{M+1} and v_{M+2} templates, respectively.

A and does not code D in the category v_{M+2} of B. Moreover, because patterns C and D represent many more features than patterns A and B, the difference between C and D is treated as noise, whereas the identical difference between A and B is considered significant. In particular, both patterns C and D are coded within the same category v_{M+3} on trials 7 and 8, and the critical feature pattern which forms the template of v_{M+3} does not contain the subpatterns at which C and D are mismatched. In contrast, these subpatterns are contained within the templates of v_{M+1} and v_{M+2} to enable these nodes to differentially classify A and B.

Figure 7 illustrates that the matching process compares whole activity patterns across a field of feature-selective cells, rather than activations of individual feature detectors, and that the properties of this matching process which enable it to stabilize network learning also automatically rescale the matching criterion. Thus the network can both differentiate finer details of simple input patterns and tolerate larger mismatches of complex input patterns. This rescaling property also defines the difference between irrelevant features and significant pattern mismatches.

If a mismatch within the attentional subsystem does not activate the orienting subsystem, then no further search for a different code occurs. Thus on trial 6 in Fig. 7, mismatched features between the template of v_{M+3} and input pattern D are treated as noise in the sense that they are rapidly suppressed in short term memory (STM) at F_1, and are eliminated from the critical feature pattern learned by the v_{M+3} template. If the mismatch does generate a search, then the mismatched features may be included in the critical feature pattern of the category to which the search leads. Thus on trial 2 of Fig. 6, the input pattern B mismatches the template of node v_{M+1}, which causes the search to select node v_{M+2}. As a result, A and B are coded by the distinct categories v_{M+1} and v_{M+2}, respectively. If a template mismatches a simple input pattern at just a few features, a search may be elicited,

thereby enabling the network to learn fine discriminations among patterns composed of few features, such as A and B. On the other hand, if a template mismatches the same number of features within a complex input pattern, then a search may not be elicited and the mismatched features may be suppressed as noise, as in the template of v_{M+3}. Thus the pattern matching process of the model automatically exhibits properties that are akin to attentional focussing, or "zooming in."

10. VIGILANCE LEVEL TUNES CATEGORICAL COARSENESS: DISCONFIRMING FEEDBACK

The previous section showed how, given each fixed vigilance level, the network automatically rescales its sensitivity to patterns of variable complexity. The present section shows that changes in the vigilance level can regulate the coarseness of the categories that are learned in response to a fixed sequence of input patterns. First we need to define the vigilance parameter ρ.

Let $|I|$ denote the number of input pathways which receive positive inputs when I is presented. Assume that each such input pathway sends an excitatory signal of fixed size P to A whenever I is presented, so that the total excitatory input to A is $P|I|$. Assume also that each F_1 node whose activity becomes positive due to I generates an inhibitory signal of fixed size Q to A, and denote by $|X|$ the number of active pathways from F_1 to A that are activated by the F_1 activity pattern X. Then the total inhibitory input from F_1 to A is $Q|X|$. When

$$P|I| > Q|X|, \tag{2}$$

the orienting subsystem A receives a net excitatory signal and generates a nonspecific reset signal to F_2 (Fig. 4c). The quantity

$$\rho \equiv \frac{P}{Q} \tag{3}$$

is called the *vigilance parameter* of A. By (2) and (3), STM reset is initiated when

$$\rho > \frac{|X|}{|I|}. \tag{4}$$

STM reset is prevented when

$$\rho \leq \frac{|X|}{|I|}. \tag{5}$$

In other words, the proportion $|X|/|I|$ of the input pattern I which is matched by the top–down template to generate X must exceed ρ in order to prevent STM reset at F_2.

While F_2 is inactive (Fig. 5b), $|X| = |I|$. Activation of A is always forbidden in this case to prevent an input I from resetting its correct F_2 code. By (5), this

constraint is achieved if

$$\rho \leq 1; \tag{6}$$

that is, if $P \leq Q$.

In summary, due to the $\frac{2}{3}$ Rule, a bad mismatch at F_1 causes a large collapse of total F_1 activity, which leads to activation of A. In order for this to happen, the system maintains a measure of the original level of total F_1 activity and compares this criterion level with the collapsed level of total F_1 activity. The criterion level is computed by summing bottom–up inputs from I to A. This sum provides a stable criterion because it is proportional to the initial activation of F_1 by the bottom–up input, and it remains unchanged as the matching process unfolds in real-time.

We now illustrate how a low vigilance level leads to learning of coarse categories, whereas a high vigilance level leads to learning of fine categories. Suppose, for example, that a low vigilance level has led to a learned grouping of inputs which need to be distinguished for successful adaptation to a prescribed input environment, but that a punishing event occurs as a consequence of this erroneous grouping (Sect. 2D). Suppose that, in addition to its negative reinforcing effects, the punishing event also has the cognitive effect of increasing sensitivity to pattern mismatches. Such an increase in sensitivity is modelled within the network by an increase in the vigilance parameter, ρ, defined by (3). Increasing this single parameter enables the network to discriminate patterns which previously were lumped together. Once these patterns are coded by different categories in F_2, the different categories can be associated with different behavioral responses. In this way, environmental feedback can enable the network to parse more finely whatever input patterns happen to occur without altering the feature detection process per se. The vigilance parameter is increased if a punishing event amplifies all the signals from the input pattern to A so that parameter P increases. Alternatively, ρ may be increased either by a nonspecific decrease in the size Q of signals from F_1 to A, or by direct input signals to A.

Figure 8 describes a series of simulations in which four input patterns—A, B, C, D—are coded. In these simulations, $A \subset B \subset C \subset D$. The different parts of the figure show how categorical learning changes with changes of ρ. When $\rho = 0.8$ (Fig. 8a), 4 categories are learned: $(A)(B)(C)(D)$. When $\rho = 0.7$ (Fig. 8b), 3 categories are learned: $(A)(B)(C, D)$. When $\rho = 0.6$ (Fig. 8c), 3 different categories are learned: $(A)(B, C)(D)$. When $\rho = 0.5$ (Fig. 8d), 2 categories are learned: $(A, B)(C, D)$. When $\rho = 0.3$ (Fig. 8e), 2 different categories are learned: $(A, B, C)(D)$. When $\rho = 0.2$ (Fig. 8f), all the patterns are lumped together into a single category.

11. RAPID CLASSIFICATION OF AN ARBITRARY TYPE FONT

In order to illustrate how an ART network codifies a more complex series of patterns, we show in Fig. 9 the first 20 trials of a simulation using alphabet letters as input patterns. In Fig. 9a, the vigilance parameter $\rho = 0.5$. In Fig. 9b, $\rho = 0.8$. Three properties are notable in these simulations. First, choosing a different vigilance parameter can determine different coding histories, such that higher vigilance induces coding into finer categories. Second, the network modifies its search order on each trial to reflect the cumulative effects of prior learning, and

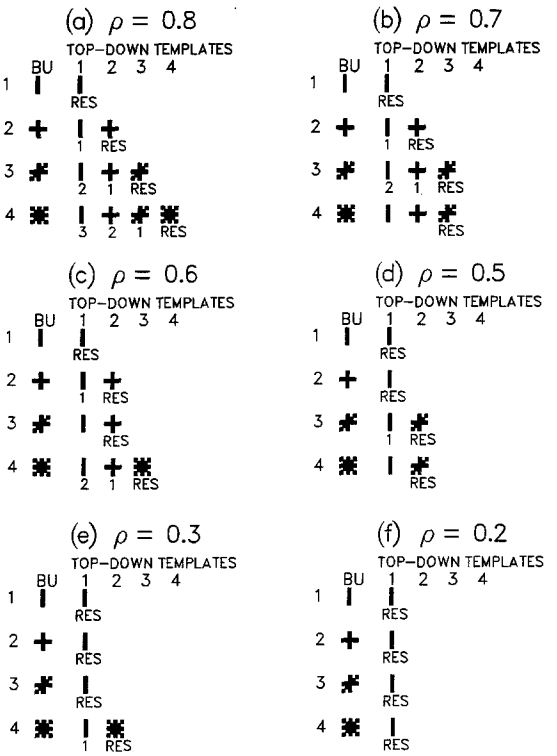

FIG. 8. Influence of vigilance level on categorical groupings: As the vigilance parameter ρ decreases, the number of categories progressively decreases.

bypasses the orienting subsystem to directly access categories after learning has taken place. Third, the templates of coarser categories tend to be more abstract because they must approximately match a larger number of input pattern exemplars.

Given $\rho = 0.5$, the network groups the 26 letter patterns into 8 stable categories within 3 presentations. In this simulation, F_2 contains 15 nodes. Thus 7 nodes remain uncoded because the network self-stabilizes its learning after satisfying criteria of vigilance and global self-consistency. Given $\rho = 0.8$ and 15 F_2 nodes, the network groups 25 of the 26 letters into 15 stable categories within 3 presentations. The 26th letter is rejected by the network in order to self-stabilize its learning while satisfying its criteria of vigilance and global self-consistency. Given a choice of ρ closer to 1, the network classifies 15 letters into 15 distinct categories within 2 presentations. In general, if an ART network is endowed with sufficiently many nodes in F_1 and F_2, it is capable of self-organizing an arbitrary ordering of arbitrarily many and arbitrarily complex input patterns into self-stabilizing recognition categories subject to the constraints of vigilance and global code self-consistency.

We now turn to a mathematical analysis of the properties which control learning and recognition by an ART network.

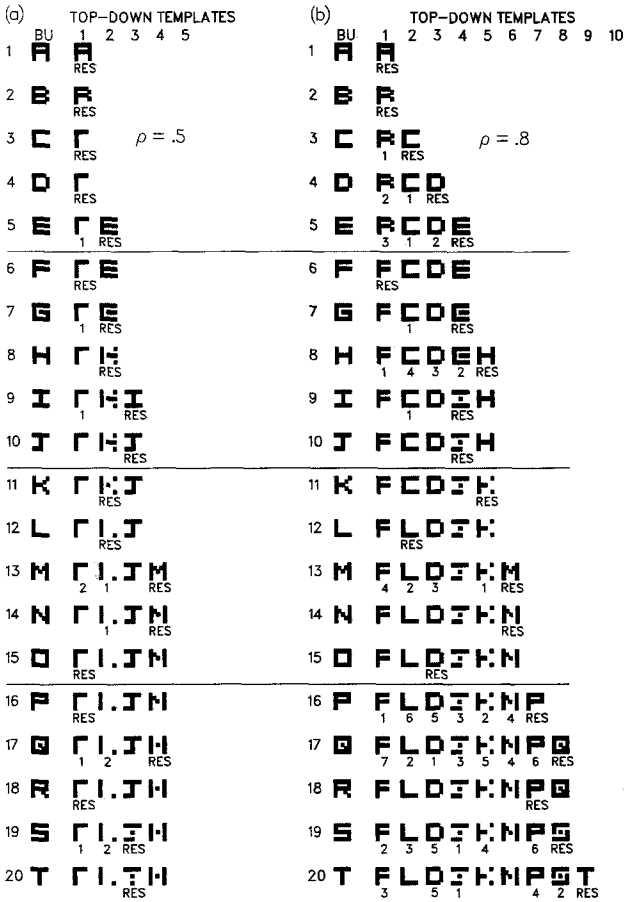

FIG. 9. Alphabet learning: different vigilance levels cause different numbers of letter categories and different critical feature patterns, or templates, to form.

12. NETWORK EQUATIONS: INTERACTIONS BETWEEN SHORT TERM MEMORY AND LONG TERM MEMORY PATTERNS

The STM and LTM equations are described below in dimensionless form [29], where the number of parameters is reduced to a minimum.

A. STM *Equations*

The STM activity x_k of any node v_k in F_1 or F_2 obeys a membrane equation of the form

$$\varepsilon \frac{d}{dt} x_k = -x_k + (1 - Ax_k) J_k^+ - (B + Cx_k) J_k^-, \tag{7}$$

where J_k^+ is the total excitatory input to v_k, J_k^- is the total inhibitory input to v_k, and all the parameters are nonnegative. If $A > 0$ and $C > 0$, then the STM activity $x_k(t)$ remains within the finite interval $[-BC^{-1}, A^{-1}]$ no matter how large the nonnegative inputs J_k^+ and J_k^- become.

We denote nodes in F_1 by v_i, where $i = 1, 2, \ldots, M$. We denote nodes in F_2 by v_j, where $j = M + 1, M + 2, \ldots, N$. Thus by (7),

$$\varepsilon \frac{d}{dt} x_i = -x_i + (1 - A_1 x_i) J_i^+ - (B_1 + C_1 x_i) J_i^- \tag{8}$$

and

$$\varepsilon \frac{d}{dt} x_j = -x_j + (1 - A_2 x_j) J_j^+ - (B_2 + C_2 x_j) J_j^- . \tag{9}$$

In the notation of (1) and Fig. 4a, the F_1 activity pattern $X = (x_1, x_2, \ldots, x_M)$ and the F_2 activity pattern $Y = (x_{M+1}, x_{M+2}, \ldots, x_N)$.

The input J_i^+ to the ith node v_i of F_1 is a sum of the bottom–up input I_i and the top–down template input V_i

$$V_i = D_1 \sum_j f(x_j) z_{ji}; \tag{10}$$

that is,

$$J_i^+ = I_i + V_i, \tag{11}$$

where $f(x_j)$ is the signal generated by activity x_j of v_j, and z_{ji} is the LTM trace in the top–down pathway from v_j to v_i. In the notation of Fig. 4b, the input pattern $I = (I_1, I_2, \ldots, I_M)$, the signal pattern $U = (f(x_{M+1}), f(x_{M+2}), \ldots, f(x_N))$, and the template pattern $V = (V_1, V_2, \ldots, V_M)$.

The inhibitory input J_i^- governs the attentional gain control signal

$$J_i^- = \sum_j f(x_j). \tag{12}$$

Thus $J_i^- = 0$ if and only if F_2 is inactive. When F_2 is active, $J_i^- > 0$ and hence term J_i^- in (8) has a nonspecific inhibitory effect on all the STM activities x_i of F_1. In Fig. 5c, this nonspecific inhibitory effect is mediated by inhibition of an active excitatory gain control channel. Such a mechanism is formally described by (12). The attentional gain control signal can be implemented in any of several formally equivalent ways. See the Appendix for some alternative systems.

The inputs and parameters of STM activities in F_2 are chosen so that the F_2 node which receives the largest input from F_1 wins the competition for STM activity. Theorems provide a basis for choosing these parameters [30–32]. The inputs J_j^+ and J_j^- to the F_2 node v_j have the following form.

Input J_j^+ adds a positive feedback signal $g(x_j)$ from v_j to itself to the bottom–up adaptive filter input T_j, where

$$T_j = D_2 \sum_i h(x_i) z_{ij}. \tag{13}$$

That is,

$$J_j^+ = g(x_j) + T_j, \tag{14}$$

where $h(x_i)$ is the signal emitted by the F_1 node v_i and z_{ij} is the LTM trace in the pathway from v_i to v_j. Input J_j^- adds up negative feedback signals $g(x_k)$ from all the other nodes in F_2,

$$J_j^- = \sum_{k \neq j} g(x_k). \tag{15}$$

In the notation of (1) and Fig. 4a, the output pattern $S = (h(x_1), h(x_2), \ldots, h(x_M))$ and the input pattern $T = (T_{M+1}, T_{M+2}, \ldots, T_N)$.

Taken together, the positive feedback signal $g(x_j)$ in (14) and the negative feedback signal J_j^- in (15) define an on-center off-surround feedback interaction which contrast-enhances the STM activity pattern Y of F_2 in response to the input pattern T. When F_2's parameters are chosen properly, this contrast-enhancement process enables F_2 to choose for STM activation only the node v_j which receives the largest input T_j. In particular, when parameter ε is small in Eq. (9), F_2 behaves approximately like a binary switching, or choice, circuit:

$$f(x_j) = \begin{cases} 1 & \text{if } T_j = \max\{T_k\} \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

In the choice case, the top–down template in (10) obeys

$$V_i = \begin{cases} D_1 z_{ji} & \text{if the } F_2 \text{ node } v_j \text{ is active} \\ 0 & \text{if } F_2 \text{ is inactive.} \end{cases} \tag{17}$$

Since V_i is proportional to the LTM trace z_{ji} of the active F_2 node v_j, we can define the template pattern that is read-out by each active F_2 node v_j to be $V^{(j)} \equiv D_1(z_{j1}, z_{j2}, \ldots, z_{jM})$.

B. LTM *Equations*

The equations for the bottom–up LTM traces z_{ij} and the top–down LTM traces z_{ji} between pairs of nodes v_i in F_1 and v_j in F_2 are formally summarized in this section to facilitate the description of how these equations help to generate useful learning and recognition properties.

The LTM trace of the bottom–up pathway from v_i to v_j obeys a learning equation of the form

$$\frac{d}{dt} z_{ij} = K_1 f(x_j)\left[-E_{ij} z_{ij} + h(x_i)\right]. \tag{18}$$

In (18), term $f(x_j)$ is a postsynaptic sampling, or learning, signal because $f(x_j) = 0$ implies $(d/dt)z_{ij} = 0$. Term $f(x_j)$ is also the output signal of v_j to pathways from v_j to F_1, as in (10).

The LTM trace of the top–down pathway from v_j to v_i also obeys a learning equation of the form

$$\frac{d}{dt} z_{ji} = K_2 f(x_j)\left[-E_{ji} z_{ji} + h(x_i)\right]. \tag{19}$$

In the present model, the simplest choice of K_2 and E_{ji} was made for the top–down LTM traces

$$K_2 = E_{ji} = 1. \tag{20}$$

A more complex choice of E_{ij} was made for the bottom–up LTM traces in order to generate the Weber Law Rule of Section 14. The Weber Law Rule requires that the positive bottom–up LTM traces learned during the encoding of an F_1 pattern X with a smaller number $|X|$ of active nodes be larger than the LTM traces learned during the encoding of an F_1 pattern with a larger number of active nodes, other things being equal. This inverse relationship between pattern complexity and bottom–up LTM trace strength can be realized by allowing the bottom–up LTM traces at each node v_j to compete among themselves for synaptic sites. The Weber Law Rule can also be generated by the STM dynamics of F_1 when competitive interactions are assumed to occur among the nodes of F_1. Generating the Weber Law Rule at F_1 rather than at the bottom–up LTM traces enjoys several advantages, and this model will be developed elsewhere [33]. In particular, implementing the Weber Law Rule at F_1 enables us to choose $E_{ij} = 1$.

Competition among the LTM traces which abut the node v_j is modelled herein by defining

$$E_{ij} = h(x_i) + L^{-1} \sum_{k \neq i} h(x_k) \tag{21}$$

and letting $K_1 =$ constant. It is convenient to write K_1 in the form $K_1 = KL$. A physical interpretation of this choice can be seen by rewriting (18) in the form

$$\frac{d}{dt} z_{ij} = Kf(x_j)\left[(1 - z_{ij})Lh(x_i) - z_{ij} \sum_{k \neq i} h(x_k)\right]. \tag{22}$$

By (22), when a postsynaptic signal $f(x_j)$ is positive, a positive presynaptic signal from the F_1 node v_i can commit receptor sites to the LTM process z_{ij} at a rate $(1 - z_{ij})Lh(x_i)Kf(x_j)$. In other words, uncommited sites—which number $(1 - z_{ij})$ out of the total population size 1—are committed by the joint action of signals $Lh(x_i)$ and $Kf(x_j)$. Simultaneously signals $h(x_k)$, $k \neq i$, which reach v_j at different patches of the v_j membrane, compete for the sites which are already committed to z_{ij} via the mass action competitive terms $-z_{ij}h(x_k)Kf(x_j)$. In other words, sites which are committed to z_{ij} lose their commitment at a rate $-z_{ij}\Sigma_{k \neq i}h(x_k)Kf(x_j)$ which is proportional to the number of committed sites z_{ij}, the total competitive input $-\Sigma_{k \neq i}h(x_k)$, and the postsynaptic gating signal $Kf(x_j)$.

Malsburg and Willshaw [34] have used a different type of competition among LTM traces in their model of retinotectal development. Translated to the present notation, Malsburg and Willshaw postulate that for each fixed F_1 node v_i, competition occurs among all the bottom–up LTM traces z_{ij} in pathways emanating from v_i in such a way as to keep the total synaptic strength $\Sigma_j z_{ij}$ constant through time. This model does not generate the Weber Law Rule. We show in Section 14 that the Weber Law Rule is essential for achieving direct access to learned categories of arbitrary input patterns in the present model.

C. STM *Reset System*

A simple type of mismatch-mediated activation of A and STM reset of F_2 by A were implemented in the simulations. As outlined in Section 10, each active input pathway sends an excitatory signal of size P to the orienting subsystem A. Potentials x_i of F_1 which exceed zero generate an inhibitory signal of size Q to A. These constraints lead to the following Reset Rule.

Reset Rule

Population A generates a nonspecific reset wave to F_2 whenever

$$\frac{|X|}{|I|} < \rho = \frac{P}{Q}, \tag{23}$$

where I is the current input pattern and $|X|$ is the number of nodes across F_1 such that $x_i > 0$. The nonspecific reset wave successively shuts off active F_2 nodes until the search ends or the input pattern I shuts off. Thus (16) must be modified as follows to maintain inhibition of all F_2 nodes which have been reset by A during the presentation of I:

F_2 Choice and Search

$$f(x_j) = \begin{cases} 1 & \text{if } T_j = \max\{T_k\colon k \in \mathbf{J}\} \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

where $\mathbf{J}$ is the set of indices of F_2 nodes which have not yet been reset on the present learning trial. At the beginning of each new learning trial, $\mathbf{J}$ is reset at $\{M+1, \ldots, N\}$. (See Fig. 1.) As a learning trial proceeds, $\mathbf{J}$ loses one index at a time until the mismatch-mediated search for F_2 nodes terminates.

13. DIRECT ACCESS TO SUBSET AND SUPERSET PATTERNS

The need for a Weber Law Rule can be motivated as follows. Suppose that a bottom–up input pattern $I^{(1)}$ activates a network in which pattern $I^{(1)}$ is perfectly coded by the adaptive filter from F_1 to F_2. Suppose that another pattern $I^{(2)}$ is also perfectly coded and that $I^{(2)}$ contains $I^{(1)}$ as a subset; that is, $I^{(2)}$ equals $I^{(1)}$ at all the nodes where $I^{(1)}$ is positive. If $I^{(1)}$ and $I^{(2)}$ are sufficiently different, they should have access to distinct categories at F_2. However, since $I^{(2)}$ equals $I^{(1)}$ at their intersection, and since all the F_1 nodes where $I^{(2)}$ does not equal $I^{(1)}$ are inactive when $I^{(1)}$ is presented, how does the network decide between the two categories when $I^{(1)}$ is presented?

To accomplish this, the node $v^{(1)}$ in F_2 which codes $I^{(1)}$ should receive a bigger signal from the adaptive filter than the node $v^{(2)}$ in F_2 which codes a superset $I^{(2)}$ of $I^{(1)}$. In order to realize this constraint, the LTM traces at $v^{(2)}$ which filter $I^{(1)}$ should be smaller than the LTM traces at $v^{(1)}$ which filter $I^{(1)}$. Since the LTM traces at $v^{(2)}$ were coded by the superset pattern $I^{(2)}$, this constraint suggests that larger patterns are encoded by smaller LTM traces. Thus the absolute sizes of the LTM traces projecting to the different nodes $v^{(1)}$ and $v^{(2)}$ reflect the overall scale of the patterns $I^{(1)}$ and $I^{(2)}$ coded by the nodes. The quantitative realization of this inverse relationship between LTM size and input pattern scale is called the Weber Law Rule.

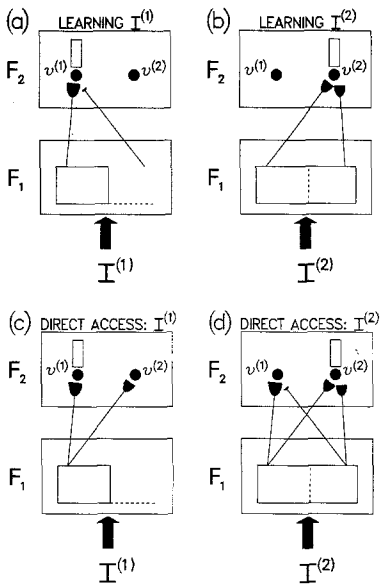

FIG. 10. The Weber Law Rule and the Associative Decay Rule enable both subset and superset input patterns to directly access distinct F_2 nodes: (a) and (b) schematize the learning induced by presentation of $I^{(1)}$ (a subset pattern) and $I^{(2)}$ (a superset pattern). Larger path endings designate larger learned LTM traces. (c) and (d) schematize how $I^{(1)}$ and $I^{(2)}$ directly access the F_2 nodes $v^{(1)}$ and $v^{(2)}$, respectively. This property illustrates how distinct, but otherwise arbitrary, input patterns can directly access different categories. No restrictions on input orthogonality or linear predictability are needed.

This inverse relationship suggests how a subset $I^{(1)}$ may selectively activate its node $v^{(1)}$ rather than the node $v^{(2)}$ corresponding to a superset $I^{(2)}$. On the other hand, the superset $I^{(2)}$ must also be able to directly activate its node $v^{(2)}$ rather than the node $v^{(1)}$ of a subset $I^{(1)}$. To achieve subset access, the positive LTM traces of $v^{(1)}$ become larger than the positive LTM traces of $v^{(2)}$. Since presentation of $I^{(2)}$ activates the entire subset pattern $I^{(1)}$, a further property is needed to understand why the subset node $v^{(1)}$ is not activated by the superset $I^{(2)}$. This property—which we call the Associative Decay Rule—implies that some LTM traces decay toward zero during learning. Thus the associative learning laws considered herein violate Hebb's [35] learning postulate.

In particular, the relative sizes of the LTM traces projecting to an F_2 node reflect the internal structuring of the input patterns coded by that node. During learning of $I^{(1)}$, the LTM traces decay toward zero in pathways which project to $v^{(1)}$ from F_1 cells where $I^{(1)}$ equals zero (Fig. 10a). Simultaneously, the LTM traces become large in the pathways which project to $v^{(1)}$ from F_1 cells where $I^{(1)}$ is positive (Fig. 10a). In contrast, during learning of $I^{(2)}$, the LTM traces become large in all the pathways which project to $v^{(2)}$ from F_1 cells where $I^{(2)}$ is positive (Fig. 10b), including those cells where $I^{(1)}$ equals zero. Since $I^{(2)}$ is a superset of $I^{(1)}$, the Weber Law Rule implies that LTM traces in pathways to $v^{(2)}$ (Fig. 10b) do not grow as large as LTM traces in pathways to $v^{(1)}$ (Fig. 10a). On the other hand, after learning occurs, more positive LTM traces exist in pathways to $v^{(2)}$ than to $v^{(1)}$. Thus a trade-off exists between the individual sizes of LTM traces and the number of positive LTM traces

which lead to each F_2 node. This trade-off enables $I^{(1)}$ to access $v^{(1)}$ (Fig. 10c) and $I^{(2)}$ to access $v^{(2)}$ (Fig. 10d).

14. WEBER LAW RULE AND ASSOCIATIVE DECAY RULE FOR BOTTOM–UP LTM TRACES

We now describe more precisely how the conjoint action of a Weber Law Rule and an Associative Decay Rule allow direct access to both subset and superset F_2 codes. To fix ideas, suppose that each input pattern I to F_1 is a pattern of 0's and 1's. Let $|I|$ denote the number of 1's in the input pattern I. The two rules can be summarized as follows.

Associative Decay Rule

As learning of I takes place, LTM traces in the bottom–up coding pathways and the top–down template pathways between an inactive F_1 node and an active F_2 node approach 0. Associative learning within the LTM traces can thus cause decreases as well as increases in the sizes of the traces. This is a non-Hebbian form of associative learning.

Weber Law Rule

As learning of I takes place, LTM traces in the bottom–up coding pathways which join active F_1 and F_2 nodes approach an asymptote of the form

$$\frac{\alpha}{\beta + |I|}, \tag{25}$$

where α and β are positive constants. By (25), larger $|I|$ values imply smaller positive LTM traces in the pathways encoding I.

Direct access by the subset $I^{(1)}$ and the superset $I^{(2)}$ can now be understood as follows. By (25), the positive LTM traces which code $I^{(1)}$ have size

$$\frac{\alpha}{\beta + |I^{(1)}|} \tag{26}$$

and the positive LTM traces which code $I^{(2)}$ have size

$$\frac{\alpha}{\beta + |I^{(2)}|}, \tag{27}$$

where $|I^{(1)}| < |I^{(2)}|$. When $I^{(1)}$ is presented at F_1, $|I^{(1)}|$ nodes in F_1 are supra-threshold. Thus the *total* input to $v^{(1)}$ is proportional to

$$T_{11} = \frac{\alpha |I^{(1)}|}{\beta + |I^{(1)}|} \tag{28}$$

and the *total* input to $v^{(2)}$ is proportional to

$$T_{12} = \frac{\alpha |I^{(1)}|}{\beta + |I^{(2)}|}. \tag{29}$$

Because (25) defines a *decreasing* function of $|I|$ and because $|I^{(1)}| < |I^{(2)}|$, it follows that $T_{11} > T_{12}$. Thus $I^{(1)}$ activates $v^{(1)}$ instead of $v^{(2)}$.

When $I^{(2)}$ is presented at F_1, $|I^{(2)}|$ nodes in F_1 are suprathreshold. Thus the *total* input to $v^{(2)}$ is proportional to

$$T_{22} = \frac{\alpha|I^{(2)}|}{\beta + |I^{(2)}|}. \tag{30}$$

We now invoke the Associative Decay Rule. Because $I^{(2)}$ is superset of $I^{(1)}$, only those F_1 nodes in $I^{(2)}$ that are also activated by $I^{(1)}$ project to positive LTM traces at $v^{(1)}$. Thus the *total* input to $v^{(1)}$ is proportional to

$$T_{21} = \frac{\alpha|I^{(1)}|}{\beta + |I^{(1)}|}. \tag{31}$$

Both T_{22} and T_{21} are expressed in terms of the Weber function

$$W(|I|) = \frac{\alpha|I|}{\beta + |I|}, \tag{32}$$

which is an *increasing* function of $|I|$. Since $|I^{(1)}| < |I^{(2)}|$, $T_{22} > T_{21}$. Thus the superset $I^{(2)}$ activates its node $v^{(2)}$ rather than the subset node $v^{(1)}$. In summary, direct access to subsets and supersets can be traced to the opposite monotonic behavior of the functions (25) and (32).

It remains to show how the Associative Decay Rule and the Weber Law Rule are generated by the STM and LTM laws (8)–(22). The Associative Decay Rule for bottom–up LTM traces follows from (22). When the F_1 node v_i is inactive, $h(x_i) = 0$. When the F_2 node v_j is active, $f(x_j) = 1$. Thus if z_{ij} is the LTM trace in a bottom–up pathway from an inactive F_1 node v_i to an active F_2 node v_j, (22) reduces to

$$\frac{d}{dt}z_{ij} = -Kz_{ij}\sum_{k\neq i} h(x_k). \tag{33}$$

The signal function $h(x_k)$ is scaled to rise steeply from 0 to the constant 1 when x_k exceeds zero. For simplicity, suppose that

$$h(x_k) = \begin{cases} 1 & \text{if } x_k > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{34}$$

Thus during a learning trial when v_i is inactive,

$$\sum_{k\neq i} h(x_k) = |X|, \tag{35}$$

where $|X|$ is the number of positive activities in the F_1 activity pattern X. By (33)

and (35), when v_i is inactive and v_j is active,

$$\frac{d}{dt}z_{ij} = -Kz_{ij}|X| \tag{36}$$

which shows that z_{ij} decays exponentially toward zero.

The Weber Law Rule for bottom–up LTM traces z_{ij} follows from (22), (24), and (34). Consider an input pattern I of 0's and 1's that activates $|I|$ nodes in F_1 and node v_j in F_2. Then, by (34),

$$\sum_{k=1}^{M} h(x_k) = |I|. \tag{37}$$

For each z_{ij} in a bottom–up pathway from an active F_1 node v_i to an active F_2 node v_j, $f(x_j) = 1$ and $h(x_i) = 1$, so

$$\frac{d}{dt}z_{ij} = K\left[(1 - z_{ij})L - z_{ij}(|I| - 1)\right]. \tag{38}$$

At equilibrium, $dz_{ij}/dt = 0$. It then follows from (38) that at equilibrium

$$z_{ij} = \frac{\alpha}{\beta + |I|} \tag{39}$$

as in (25), with $\alpha = L$ and $\beta = L - 1$. Both α and β must be positive, which is the case if $L > 1$. By (22), this means that each lateral inhibitory signal $-h(x_k)$, $k \neq i$, is weaker than the direct excitatory signal $Lh(x_i)$, other things being equal.

When top–down signals from F_2 to F_1 supplement a bottom–up input pattern I to F_1, the number $|X|$ of positive activities in X may become smaller than $|I|$ due to the $\frac{2}{3}$ Rule. If v_i remains active after the F_2 node v_j becomes active, (38) generalizes to

$$\frac{d}{dt}z_{ij} = K\left[(1 - z_{ij})L - z_{ij}(|X| - 1)\right]. \tag{40}$$

By combining (36) and (40), both the Associative Decay Rule and the Weber Law Rule for bottom–up LTM traces may be understood as consequences of the LTM equation

$$\frac{d}{dt}z_{ij} = \begin{cases} K\left[(1 - z_{ij})L - z_{ij}(|X| - 1)\right] & \text{if } v_i \text{ and } v_j \text{ are active} \\ -K|X|z_{ij} & \text{if } v_i \text{ is inactive and } v_j \text{ is active} \\ 0 & \text{if } v_j \text{ is inactive.} \end{cases} \tag{41}$$

Evaluation of term $|X|$ in (41) depends upon whether or not a top–down template perturbs F_1 when a bottom–up input pattern I is active.

15. TEMPLATE LEARNING RULE AND ASSOCIATIVE DECAY RULE FOR TOP–DOWN LTM TRACES

The Template Learning Rule and the Associative Decay Rule together imply that the top–down LTM traces in all the pathways from an F_2 node v_j encode the critical feature pattern of all input patterns which have activated v_j without triggering F_2 reset. To see this, as in Section 14, suppose that an input pattern I of 0's and 1's is being learned.

Template Learning Rule

As learning of I takes place, LTM traces in the top–down pathways from an active F_2 node to an active F_1 node approach 1.

The Template Learning Rule and the Associative Decay Rule for top–down LTM traces z_{ji} follow by combining (19) and (20) to obtain

$$\frac{d}{dt}z_{ji} = f(x_j)\left[-z_{ji} + h(x_i)\right]. \tag{42}$$

If the F_2 node v_j is active and the F_1 node v_i is inactive, then $h(x_i) = 0$ and $f(x_j) = 1$, so (42) reduces to

$$\frac{d}{dt}z_{ji} = -z_{ji}. \tag{43}$$

Thus z_{ji} decays exponentially toward zero and the Associative Decay Rule holds. On the other hand, if both v_i and v_j are active, then $f(x_j) = h(x_i) = 1$, so (42) reduces to

$$\frac{d}{dt}z_{ji} = -z_{ji} + 1. \tag{44}$$

Thus z_{ji} increases exponentially toward 1 and the Template Learning Rule holds.

Combining equations (42)–(44) leads to the learning rule governing the LTM traces z_{ji} in a top–down template

$$\frac{d}{dt}z_{ji} = \begin{cases} -z_{ji} + 1 & \text{if } v_i \text{ and } v_j \text{ are active} \\ -z_{ji} & \text{if } v_i \text{ is inactive and } v_j \text{ is active} \\ 0 & \text{if } v_j \text{ is inactive.} \end{cases} \tag{45}$$

Equation (45) says that the template of v_j tries to learn the activity pattern across F_1 when v_j is active.

The $\frac{2}{3}$ Rule controls which nodes v_i in (45) remain active in response to an input pattern I. The $\frac{2}{3}$ Rule implies that if the F_2 node v_j becomes active while the F_1 node v_i is receiving a large bottom–up input I_i, then v_i will remain active only if z_{ji} is sufficiently large. Hence there is some critical strength of the top–down LTM traces such that if z_{ji} falls below that strength, then v_i will never again be active when v_j is active, even if I_i is large. As long as z_{ji} remains above the critical LTM strength, it will increase when I_i is large and v_j is active, and decrease when I_i is

small and v_j is active. Once z_{ji} falls below the critical LTM strength, it will decay toward 0 whenever v_j is active; that is, the feature represented by v_i drops out of the critical feature pattern encoded by v_j.

These and related properties of the network can be summarized compactly using the following notation.

Let **I** denote the set of indices of nodes v_i which receive a positive input from the pattern I. When I is a pattern of 0's and 1's, then

$$I_i = \begin{cases} 1 & \text{if } i \in \mathbf{I} \\ 0 & \text{otherwise,} \end{cases} \tag{46}$$

where **I** is a subset of the F_1 index set $\{1 \ldots M\}$. As in Section 12, let $V^{(j)} = D_1(z_{j1} \cdots z_{ji} \cdots z_{jM})$ denote the template pattern of top–down LTM traces in pathways leading from the F_2 node v_j. The index set $\mathbf{V}^{(j)} = \mathbf{V}^{(j)}(t)$ is defined as follows: $i \in \mathbf{V}^{(j)}$ iff z_{ji} is larger than the critical LTM strength required for v_i to be active when v_j is active and $i \in \mathbf{I}$. For fixed t, let **X** denote the subset of indices $\{1 \ldots M\}$ such that $i \in \mathbf{X}$ iff the F_1 node v_i is active at time t.

With this notation, the $\frac{2}{3}$ Rule can be summarized by stating that when a pattern I is presented,

$$\mathbf{X} = \begin{cases} \mathbf{I} & \text{if } F_2 \text{ is inactive} \\ \mathbf{I} \cap \mathbf{V}^{(j)} & \text{if the } F_2 \text{ node } v_j \text{ is active.} \end{cases} \tag{47}$$

The link between STM dynamics at F_1 and F_2 and LTM dynamics between F_1 and F_2 can now be succinctly expressed in terms of (47),

$$\frac{d}{dt} z_{ij} = \begin{cases} K\left[(1 - z_{ij})L - z_{ij}(|\mathbf{X}| - 1)\right] & \text{if } i \in \mathbf{X} \text{ and } f(x_j) = 1 \\ -K|\mathbf{X}|z_{ij} & \text{if } i \notin \mathbf{X} \text{ and } f(x_j) = 1 \\ 0 & \text{if } f(x_j) = 0 \end{cases} \tag{48}$$

and

$$\frac{d}{dt} z_{ji} = \begin{cases} -z_{ji} + 1 & \text{if } i \in \mathbf{X} \text{ and } f(x_j) = 1 \\ -z_{ji} & \text{if } i \notin \mathbf{X} \text{ and } f(x_j) = 1 \\ 0 & \text{if } f(x_j) = 0. \end{cases} \tag{49}$$

A number of definitions that were made intuitively in Sections 3–9 can now be summarized as follows.

Definitions

Coding

An active F_2 node v_J is said to *code* an input I on a given trial if no reset of v_J occurs after the template $V^{(J)}$ is read out at F_1.

Reset could, in principle, occur due to three different factors. The read-out of the template $V^{(J)}$ can change the activity pattern X across F_1. The new pattern X could

conceivably generate a maximal input via the $F_1 \to F_2$ adaptive filter to an F_2 node other than v_J. The theorems below show how the $\frac{2}{3}$ Rule and the learning rules prevent template read-out from undermining the choice of v_J via the $F_1 \to F_2$ adaptive filter. Reset of v_J could also, in principle, occur due to the learning induced in the LTM traces z_{iJ} and z_{Ji} by the choice of v_J. In a real-time learning system whose choices are determined by a continuous flow of bottom–up and top–down signals, one cannot take for granted that the learning process, which alters the sizes of these signals, will maintain a choice within a single learning trial. The theorems in the next sections state conditions which prevent either template readout or learning from resetting the F_2 choice via the adaptive filter from F_1 to F_2.

Only the third possible reset mechanism—activation of the orienting subsystem A by a mismatch at F_1—is allowed to reset the F_2 choice. Equations (5) and (47) imply that if v_J becomes active during the presentation of I, then inequality

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \geq \rho|\mathbf{I}| \tag{50}$$

is a necessary condition to prevent reset of v_J by activation of A. Sufficient conditions are stated in the theorems below.

Direct Access

Pattern I is said to have *direct access* to an F_2 node v_J if presentation of I leads at once to activation of v_J and v_J codes I on that trial.

By Eqs. (13) and (34), input I chooses node v_J first if, for all $j \neq J$,

$$\sum_{i \in I} z_{iJ} > \sum_{i \in I} z_{ij}. \tag{51}$$

The conditions under which v_J then codes I are characterized in the theorems below.

Fast Learning

For the remainder of this article we consider the *fast learning case* in which learning rates enable LTM traces to approximately reach the asymptotes determined by the STM patterns on each trial. Given the fast learning assumption, at the end of a trial during which v_J was active, (48) implies that

$$z_{iJ} \cong \begin{cases} \dfrac{L}{L - 1 + |\mathbf{X}|} & \text{if } i \in \mathbf{X} \\ 0 & \text{if } i \notin \mathbf{X} \end{cases} \tag{52}$$

and (49) implies that

$$z_{Ji} \cong \begin{cases} 1 & \text{if } i \in \mathbf{X} \\ 0 & \text{if } i \notin \mathbf{X}. \end{cases} \tag{53}$$

Thus although $z_{ij} \neq z_{ji}$ in (52) and (53), z_{ij} is large iff z_{ji} is large and $z_{ij} = 0$ iff $z_{ji} = 0$. We can therefore introduce the following definition.

Asymptotic Learning

An F_2 node v_j has *asymptotically learned* the STM pattern X if its LTM traces z_{ij} and z_{ji} satisfy (52) and (53).

By (47), $\mathbf{X}$ in (52) and (53) equals either $\mathbf{I}$ or $\mathbf{I} \cap \mathbf{V}^{(J)}$. This observation motivates the following definition.

Perfect Learning

An F_2 node v_j has *perfectly learned* an input pattern I iff v_j has asymptotically learned the STM pattern $X = I$.

16. DIRECT ACCESS TO NODES CODING PERFECTLY LEARNED PATTERNS

We can now prove the following generalization of the fact that subset and superset nodes can be directly accessed (Sect. 13).

THEOREM 1 (Direct access by perfectly learned patterns). *An input pattern I has direct access to a node v_J which has perfectly learned I if $L > 1$ and all initial bottom–up* LTM *traces satisfy the*

$$\textit{Direct Access Inequality} \qquad 0 < z_{ij}(0) < \frac{L}{L - 1 + M}, \tag{54}$$

where M is the number of nodes in F_1.

Proof. In order to prove that I has direct access to v_J we need to show that: (i) v_J is the first F_2 node to be chosen; (ii) v_J remains the chosen node after its template $V^{(J)}$ is read out at F_1; (iii) read out of $V^{(J)}$ does not lead to F_2 reset by the orienting subsystem; and (iv) v_J remains active as fast learning occurs.

To prove property (i), we must establish that, at the start of the trial, $T_J > T_j$ for all $j \neq J$. When I is presented, $|\mathbf{I}|$ active pathways project to each F_2 node. In particular, by (13) and (34),

$$T_J = D_2 \sum_{i \in \mathbf{I}} z_{iJ} \tag{55}$$

and

$$T_j = D_2 \sum_{i \in \mathbf{I}} z_{ij}. \tag{56}$$

Because node v_J perfectly codes I at the start of the trial, it follows from (52) that

$$z_{iJ} = \begin{cases} \dfrac{L}{L - 1 + |\mathbf{I}|} & \text{if } i \in \mathbf{I} \\ 0 & \text{if } i \notin \mathbf{I}. \end{cases} \tag{57}$$

By (55) and (57),

$$T_J = \frac{D_2 L |\mathbf{I}|}{L - 1 + |\mathbf{I}|}. \tag{58}$$

In order to evaluate T_j in (56), we need to consider nodes v_j which have asymptotically learned a different pattern than I, as well as nodes v_j which are as yet uncommitted. Suppose that v_j, $j \neq J$, has asymptotically learned a pattern $V^{(j)} \neq I$.

Then by (52),

$$z_{ij} = \begin{cases} \dfrac{L}{L-1+|\mathbf{V}^{(j)}|} & \text{if } i \in \mathbf{V}^{(j)} \\ 0 & \text{if } i \notin \mathbf{V}^{(j)}. \end{cases} \tag{59}$$

By (59), the only positive LTM traces in the sum $\Sigma_{i \in \mathbf{I}} z_{ij}$ in (56) are the traces with indices $i \in \mathbf{I} \cap \mathbf{V}^{(j)}$. Moreover, all of these positive LTM traces have the same value. Thus (59) implies that

$$T_j = \frac{D_2 L |\mathbf{I} \cap \mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|}. \tag{60}$$

We now prove that T_J in (58) is larger than T_j in (60) if $L > 1$; that is,

$$\frac{|\mathbf{I}|}{L-1+|\mathbf{I}|} > \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|}. \tag{61}$$

Suppose first that $|\mathbf{V}^{(j)}| > |\mathbf{I}|$. Then $|\mathbf{I}| \geq |\mathbf{I} \cap \mathbf{V}^{(j)}|$ and $(L-1+|\mathbf{I}|) < (L-1+|\mathbf{V}^{(j)}|)$, which together imply (61).

Suppose next that $|\mathbf{V}^{(j)}| \leq |\mathbf{I}|$. Then, since $\mathbf{V}^{(j)} \neq \mathbf{I}$, it follows that $|\mathbf{I}| > |\mathbf{I} \cap \mathbf{V}^{(j)}|$. Thus, since the function $w/(L-1+w)$ is an increasing function of w,

$$\frac{|\mathbf{I}|}{L-1+|\mathbf{I}|} > \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L-1+|\mathbf{I} \cap \mathbf{V}^{(j)}|}. \tag{62}$$

Finally, since $|\mathbf{V}^{(j)}| \leq |\mathbf{I} \cap \mathbf{V}^{(j)}|$,

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L-1+|\mathbf{I} \cap \mathbf{V}^{(j)}|} \geq \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|}. \tag{63}$$

Inequalities (62) and (63) together imply (61). This completes the proof that I first activates v_J rather than any other previously coded node v_j.

It remains to prove that I activates v_J rather than an uncommitted node v_j which has not yet been chosen to learn any category. The LTM traces of each uncommitted node v_j obey the Direct Access Inequality (54), which along with $|\mathbf{I}| \leq M$ implies that

$$T_J = \frac{D_2 L |\mathbf{I}|}{L-1+|\mathbf{I}|} \geq \frac{D_2 L |\mathbf{I}|}{L-1+M} > D_2 \sum_{i \in \mathbf{I}} z_{ij} = T_j. \tag{64}$$

This completes the proof of property (i).

The proof of property (ii), that v_J remains the chosen node after its template $V^{(J)}$ is read out, follows immediately from the fact that $\mathbf{V}^{(J)} = \mathbf{I}$. By (47), the set $\mathbf{X}$ of active nodes remains equal to $\mathbf{I}$ after $V^{(J)}$ is read-out. Thus T_J and T_j are unchanged by read-out of $V^{(J)}$, which completes the proof of property (ii).

Property (iii) also follows immediately from the fact that $\mathbf{I} \cap \mathbf{V}^{(J)} = \mathbf{I}$ in the inequality

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \geq \rho|\mathbf{I}|. \tag{50}$$

Property (iv) follows from the fact that, while v_J is active, no new learning occurs, since v_J had already perfectly learned input pattern I before the trial began. This completes the proof of Theorem 1.

17. INITIAL STRENGTHS OF LTM TRACES

A. *Direct Access Inequality: Initial Bottom–Up* LTM *Traces are Small*

Theorem 1 shows that the Direct Access Inequality (54) is needed to prevent uncommitted nodes from interfering with the direct activation of perfectly coded nodes. We now show that violation of the Direct Access Inequality may force all uncommitted nodes to code a single input pattern, and thus to drastically reduce the coding capacity of F_2.

To see this, suppose that for all v_j in F_2 and all $i \in \mathbf{I}$,

$$z_{ij}(0) > \frac{L}{L - 1 + |\mathbf{I}|}. \tag{65}$$

Suppose that on the first trial, v_{j_1} is the first F_2 node to be activated by input I. Thus $T_{j_1} > T_j$, where $j \neq j_1$, at the start of the trial. While activation of v_{j_1} persists, T_{j_1} decreases towards the value $D_2 L|\mathbf{I}|(L - 1 + |\mathbf{I}|)^{-1}$ due to learning. However, for all $j \neq j_1$,

$$T_j = D_2 \sum_{i \in \mathbf{I}} z_{ij}(0) > \frac{D_2 L|\mathbf{I}|}{L - 1 + |\mathbf{I}|}. \tag{66}$$

By (66), T_{j_1} eventually decreases so much that $T_{j_1} = T_{j_2}$ for some other node v_{j_2} in F_2. Thereafter, T_{j_1} and T_{j_2} both approach $D_2 L|\mathbf{I}|(L - 1 + |\mathbf{I}|)^{-1}$ as activation alternates between v_{j_1} and v_{j_2}. Due to inequality (65), all F_2 nodes v_j eventually are activated and their T_j values decrease towards $D_2 L|\mathbf{I}|(L - 1 + |\mathbf{I}|)^{-1}$. Thus *all* the F_2 nodes asymptotically learn the same input pattern I. The Direct Access Inequality (54) prevents these anomalies from occurring. It makes precise the idea that the initial values of the bottom–up LTM traces $z_{ij}(0)$ must not be too large.

B. *Template Learning Inequality: Initial Top–Down Traces are Large*

In contrast, the initial top–down LTM traces $z_{ji}(0)$ must not be too small. The $\frac{2}{3}$ Rule implies that if the initial top–down LTM traces $z_{ji}(0)$ were too small, then no uncommitted F_2 node could ever learn any input pattern, since all F_1 activity would be quenched as soon as F_2 became active.

To understand this issue more precisely, suppose that an input I is presented. While F_2 is inactive, $\mathbf{X} = \mathbf{I}$. Suppose that, with or without a search, the uncommitted F_2 node v_J becomes active on that trial. In order for v_J to be able to encode I given an arbitrary value of the vigilance parameter ρ, it is necessary that $\mathbf{X}$ remain equal to $\mathbf{I}$ after the template $V^{(J)}$ has been read out; that is,

$$\mathbf{I} \cap \mathbf{V}^{(J)}(0) = \mathbf{I} \qquad \text{for any } I. \tag{67}$$

Because I is arbitrary, the $\frac{2}{3}$ Rule requires that $\mathbf{V}^{(J)}$ initially be the entire set $\{1,\ldots,M\}$. In other words, the initial strengths of all the top–down LTM traces $z_{J1}\ldots z_{JM}$ must be greater than the critical LTM strength, denoted by $\bar{z}$, that is required to maintain suprathreshold STM activity in each F_1 node v_i such that $i \in \mathbf{I}$. Equation (49) and the $\frac{2}{3}$ Rule then imply that, as long as I persists and v_J remains active, $z_{Ji} \to 1$ for $i \in \mathbf{I}$ and $z_{Ji} \to 0$ for $i \notin \mathbf{I}$. Thus $\mathbf{V}^{(J)}$ contracts from $\{1,\ldots,M\}$ to $\mathbf{I}$ as the node v_J encodes the pattern I.

It is shown in the Appendix that the following inequalities imply the $\frac{2}{3}$ Rule

$\frac{2}{3}$ Rule Inequalities

$$\max\{1, D_1\} < B_1 < 1 + D_1; \tag{68}$$

and that the critical top–down LTM strength is

$$\bar{z} \equiv \frac{B_1 - 1}{D_1}. \tag{69}$$

Then the

Template Learning Inequality

$$1 \geq z_{ji}(0) > \bar{z} \tag{70}$$

implies that $\mathbf{V}^{(J)}(0) = \{1\ldots M\}$ for all j, so (67) holds.

C. *Activity-Dependent Nonspecific Tuning of Initial* LTM *Values*

Equations (52) and (53) suggest a simple developmental process by which the opposing constraints on $z_{ij}(0)$ and $z_{ji}(0)$ of Sections 17A and B can be achieved. Suppose that at a developmental stage prior to the category learning stage, all F_1 and F_2 nodes become endogenously active. Let this activity nonspecifically influence F_1 and F_2 nodes for a sufficiently long time interval to allow their LTM traces to approach their asymptotic values. The presence of noise in the system implies that the initial z_{ij} and z_{ji} values are randomly distributed close to these asymptotic values. At the end of this stage, then,

$$z_{ij}(0) \cong \frac{L}{L - 1 + M} \tag{71}$$

and

$$z_{ji}(0) \cong 1 \tag{72}$$

for all $i = 1\ldots M$ and $j = M + 1\ldots N$. The bottom–up LTM traces $z_{ij}(0)$ and the top–down LTM traces $z_{ji}(0)$ are then as large as possible, and still satisfy the Direct Access Inequality (54) and the Template Learning Inequality (70). Switching from this early developmental stage to the category learning stage could then be viewed as a switch from an endogenous source of broadly-distributed activity to an exogenous source of patterned activity.

18. SUMMARY OF THE MODEL

Below, we summarize the hypotheses that define the model. All subsequent theorems in the article assume that these hypotheses hold.

Binary Input Patterns

$$I_i = \begin{cases} 1 & \text{if } i \in \mathbf{I} \\ 0 & \text{otherwise.} \end{cases} \tag{46}$$

Automatic Bottom–Up Activation and $\frac{2}{3}$ Rule

$$\mathbf{X} = \begin{cases} \mathbf{I} & \text{if } F_2 \text{ is inactive} \\ \mathbf{I} \cap \mathbf{V}^{(J)} & \text{if the } F_2 \text{ node } v_j \text{ is active.} \end{cases} \tag{47}$$

Weber Law Rule and Bottom–Up Associative Decay Rule

$$\frac{d}{dt} z_{ij} = \begin{cases} K\left[(1 - z_{ij})L - z_{ij}(|\mathbf{X}| - 1)\right] & \text{if } i \in \mathbf{X} \text{ and } f(x_j) = 1 \\ -K|\mathbf{X}|z_{ij} & \text{if } i \notin \mathbf{X} \text{ and } f(x_j) = 1 \\ 0 & \text{if } f(x_j) = 0. \end{cases} \tag{48}$$

Template Learning Rule and Top–Down Associative Decay Rule

$$\frac{d}{dt} z_{ji} = \begin{cases} -z_{ji} + 1 & \text{if } i \in \mathbf{X} \text{ and } f(x_j) = 1 \\ -z_{ji} & \text{if } i \notin \mathbf{X} \text{ and } f(x_j) = 1 \\ 0 & \text{if } f(x_j) = 0. \end{cases} \tag{49}$$

Reset Rule

An active F_2 node v_j is reset if

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{I}|} < \rho \equiv \frac{P}{Q}. \tag{73}$$

Once a node is reset, it remains inactive for the duration of the trial.

F_2 Choice and Search

If $\mathbf{J}$ is the index set of F_2 nodes which have not yet been reset on the present learning trial, then

$$f(x_j) = \begin{cases} 1 & \text{if } T_j = \max\{T_k\colon k \in \mathbf{J}\} \\ 0 & \text{otherwise,} \end{cases} \tag{24}$$

where

$$T_j = D_2 \sum_{i \in \mathbf{X}} z_{ij}. \tag{74}$$

In addition, all STM activities x_i and x_j are reset to zero after each learning trial. The initial bottom–up LTM traces $z_{ij}(0)$ are chosen to satisfy the

Direct Access Inequality

$$0 < z_{ij}(0) < \frac{L}{L - 1 + M}. \tag{54}$$

The initial top–down LTM traces are chosen to satisfy the

Template Learning Inequality

$$1 \geq z_{ji}(0) > \bar{z} \equiv \frac{B_1 - 1}{D_1}. \tag{75}$$

Fast Learning

It is assumed that fast learning occurs so that, when v_j in F_2 is active, all LTM traces approach the asymptotes,

$$z_{ij} \cong \begin{cases} \dfrac{L}{L - 1 + |\mathbf{X}|} & \text{if } i \in \mathbf{X} \\ 0 & \text{if } i \notin \mathbf{X} \end{cases} \tag{52}$$

and

$$z_{ji} \cong \begin{cases} 1 & \text{if } i \in \mathbf{X} \\ 0 & \text{if } i \notin \mathbf{X}. \end{cases} \tag{53}$$

on each learning trial. A complete listing of parameter constraints is provided in Table 1.

TABLE 1
Parameter Constraints

$A_1 \geq 0$
$C_1 \geq 0$
$\max\{1, D_1\} < B_1 < 1 + D_1$
$0 < \varepsilon \ll 1$
$K = O(1)$
$L > 1$
$0 < \rho \leq 1$
$0 < z_{ij}(0) < \dfrac{L}{L - 1 + M}$
$1 \geq z_{ji}(0) > \bar{z} \equiv \dfrac{B_1 - 1}{D_1}$
$0 \leq I_i, f, g, h \leq 1$

19. ORDER OF SEARCH AND STABLE CHOICES IN SHORT-TERM MEMORY

We will now analyze further properties of the class of ART systems which satisfy the hypotheses in Section 18. We will begin by characterizing the order of search. This analysis provides a basis for proving that learning self-stabilizes and leads to recognition by direct access.

This discussion of search order does not analyse where the search ends. Other things being equal, a network with a higher level of vigilance will require better F_1 matches, and hence will search more deeply, in response to each input pattern. The set of learned filters and templates thus depends upon the prior levels of vigilance, and the same ordering of input patterns may generate different LTM encodings due to the settings of the nonspecific vigilance parameter. The present discussion considers the order in which search will occur in response to a single input pattern which is presented after an arbitrary set of prior inputs has been asymptotically learned.

We will prove that the values of the F_2 input functions T_j at the start of each trial determine the order in which F_2 nodes are searched, assuming that no F_2 nodes are active before the trial begins. To distinguish these initial T_j values from subsequent T_j values, let O_j denote the value of T_j at the start of a trial. We will show that, if these values are ordered by decreasing size, as in

$$O_{j_1} > O_{j_2} > O_{j_3} > \cdots, \tag{76}$$

then F_2 nodes are searched in the order $v_{j_1}, v_{j_2}, v_{j_3}, \ldots$ on that trial. To prove this result, we first derive a formula for O_j.

When an input I is first presented on a trial,

$$O_j = D_2 \sum_{i \in \mathbf{I}} z_{ij}, \tag{77}$$

where the z_{ij}'s are evaluated at the start of the trial. By the Associative Decay Rule, z_{ij} in (77) is positive only if $i \in \mathbf{V}^{(j)}$, where $\mathbf{V}^{(j)}$ is also evaluated at the start of the trial. Thus by (77),

$$O_j = D_2 \sum_{i \in \mathbf{I} \cap \mathbf{V}^{(j)}} z_{ij}. \tag{78}$$

If the LTM traces z_{ij} have undergone learning on a previous trial, then (52) implies

$$z_{ij} = \frac{L}{L - 1 + |\mathbf{V}^{(j)}|} \tag{79}$$

for all $i \in \mathbf{V}^{(j)}$. If v_j is an uncommitted node, then the Template Learning Inequality implies that $\mathbf{I} \cap \mathbf{V}^{(j)} = \mathbf{I}$. Combining these facts leads to the following formula for O_j.

Order Function

$$O_j = \begin{cases} \dfrac{D_2 L |\mathbf{I} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|} & \text{if } v_j \text{ has been chosen on a previous trial} \\ D_2 \Sigma_{i \in \mathbf{I}} z_{ij}(0) & \text{if } v_j \text{ is an uncommitted node.} \end{cases} \tag{80}$$

In response to input pattern I, (76) implies that node v_{j_1} is initially chosen by F_2. After v_{j_1} is chosen, it reads-out template $V^{(j_1)}$ to F_1. When $V^{(j_1)}$ and I both perturb F_1, a new activity pattern X is registered at F_1, as in Fig. 4b. By the $\frac{2}{3}$ Rule, $\mathbf{X} = \mathbf{I} \cap \mathbf{V}^{(j_1)}$. Consequently, a new bottom–up signal pattern from F_1 to F_2 will then be registered at F_2. How can we be sure that v_{j_1} will continue to receive the largest input from F_1 after its template is processed by F_1? In other words, does read-out of the top–down template $V^{(j_1)}$ confirm the choice due to the ordering of bottom–up signals O_j in (76)? Theorem 2 provides this guarantee. Then Theorem 3 shows that the ordering of initial T_j values determines the order of search on each trial despite the fact that the T_j values can fluctuate dramatically as different F_2 nodes get activated.

THEOREM 2 (Stable choices in STM). *Assume the model hypotheses of Section* 18. *Suppose that an F_2 node v_J is chosen for* STM *storage instead of another node v_j because $O_J > O_j$. Then read-out of the top–down template $V^{(J)}$ preserves the inequality $T_J > T_j$ and thus confirms the choice of v_J by the bottom–up filter.*

Proof. Suppose that a node v_J is activated due to the input pattern I, and that v_J is not an uncommitted node. When v_J reads out the template $V^{(J)}$ to F_1, $\mathbf{X} = \mathbf{I} \cap \mathbf{V}^{(J)}$ by the $\frac{2}{3}$ Rule. Then

$$T_j = D_2 \sum_{i \in \mathbf{I} \cap \mathbf{V}^{(J)}} z_{ij}. \tag{81}$$

Since $z_{ij} > 0$ only if $i \in \mathbf{V}^{(j)}$,

$$T_j = D_2 \sum_{i \in \mathbf{I} \cap \mathbf{V}^{(J)} \cap \mathbf{V}^{(j)}} z_{ij}. \tag{82}$$

By (79), if T_j is not an uncommitted node,

$$T_j = \frac{D_2 L |\mathbf{I} \cap \mathbf{V}^{(J)} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|}. \tag{83}$$

By (80) and (83),

$$T_j \leq O_j. \tag{84}$$

Similarly, if v_j is an uncommitted node, the sum T_j in (82) is less than or equal to the sum O_j in (80). Thus read-out of template $V^{(J)}$ can only cause the bottom–up signals T_j, other than T_J, to decrease. Signal T_J, on the other hand, remains unchanged after read-out of $V^{(J)}$. This can be seen by replacing $V^{(j)}$ in (83) by $V^{(J)}$. Then

$$T_J = \frac{D_2 L |\mathbf{I} \cap \mathbf{V}^{(J)}|}{L - 1 + |\mathbf{V}^{(J)}|}. \tag{85}$$

Hence, after $V^{(J)}$ is read-out

$$T_J = O_J. \tag{86}$$

Combining (84) and (86) shows that inequality $T_J > T_j$ continues to hold after $V^{(J)}$

is read out, thereby proving that top–down template read-out confirms the F_2 choice of the bottom–up filter.

The same is true if v_J is an uncommitted node. Here, the Template Learning Inequality shows that $\mathbf{X} = \mathbf{I}$ even after $v^{(J)}$ is read out. Thus *all* bottom–up signals T_j remain unchanged after template read-out in this case. This completes the proof of Theorem 2.

Were the $\frac{2}{3}$ Rule not operative, read-out of the template $V^{(j_1)}$ might activate many F_1 nodes that had not previously been activated by the input I alone. For example, a top–down template could, in principle, activate all the nodes of F_1, thereby preventing the input pattern, as a pattern, from being coded. Alternatively, disjoint input patterns could be coded by a single node, despite the fact that these two patterns do not share any features. The $\frac{2}{3}$ Rule prevents such coding anomalies from occurring.

THEOREM 3 (Initial filter values determine search order). *The Order Function O_j determines the order of search no matter how many times F_2 is reset during a trial.*

Proof. Since $O_{j_1} > O_{j_2} > \cdots$, node v_{j_1} is the first node to be activated on a given trial. After template $V^{(j_1)}$ is read out, Theorem 2 implies that

$$T_{j_1} = O_{j_1} > \max\{O_j\colon\ j \neq j_1\} \geq \max\{T_j\colon\ j \neq j_1\}, \tag{87}$$

even though the full ordering of the T_j's may be different from that defined by the O_j's. If v_{j_1} is reset by the orienting subsystem, then template $V^{(j_1)}$ is shut off for the remainder of the trial and subsequent values of T_{j_1} do not influence which F_2 nodes will be chosen.

As soon as v_{j_1} and $V^{(j_1)}$ are shut off, $T_j = O_j$ for all $j \neq j_1$. Since $O_{j_2} > O_{j_3} > \cdots$, node v_{j_2} is chosen next and template $V^{(j_2)}$ is read-out. Theorem 2 implies that

$$T_{j_2} = O_{j_2} > \max\{O_j\colon\ j \neq j_1, j_2\} \geq \max\{T_j\colon\ j \neq j_1, j_2\}. \tag{88}$$

Thus $V^{(j_2)}$ confirms the F_2 choice due to O_{j_2} even though the ordering of T_j values may differ both from the ordering of O_j values and from the ordering of T_j values when $V^{(j_1)}$ was active.

This argument can now be iterated to show that the values $O_{j_1} > O_{j_2} > \cdots$ of the Order Function determine the order of search. This completes the proof of Theorem 3.

20. STABLE CATEGORY LEARNING

Theorems 2 and 3 describe choice and search properties which occur on such a fast time scale that no new learning can occur. We now analyse properties of learning throughout an entire trial, and use these properties to show that code learning self-stabilizes across trials in response to an arbitrary list of binary input patterns. In Theorem 2, we proved that read-out of a top–down template confirms the F_2 choice made by the bottom–up filter. In Theorem 4, we will prove that learning also confirms the F_2 choice and does not trigger reset by the orienting subsystem. In addition, learning on a single trial causes monotonic changes in the LTM traces.

THEOREM 4 (Learning on a single trial). *Assume the model hypotheses of Section* 18. *Suppose that an F_2 node v_J is chosen for* STM *storage and that read-out of the template $V^{(J)}$ does not immediately lead to reset of node v_J by the orienting subsystem. Then the* LTM *traces z_{iJ} and z_{Ji} change monotonically in such a way that T_J increases and all other T_j remain constant, thereby confirming the choice of v_J by the adaptive filter. In addition, the set $\mathbf{I} \cap \mathbf{V}^{(J)}$ remains constant during learning, so that learning does not trigger reset of v_J by the orienting subsystem.*

Proof. We first show that the LTM traces $z_{Ji}(t)$ can only change monotonically and that the set $\mathbf{X}(t)$ does not change as long as v_J remains active. These conclusions follow from the learning rules for the top–down LTM traces z_{Ji}. Using these facts, we then show that the $z_{iJ}(t)$ change monotonically, that $T_J(t)$ can only increase, and that all other $T_j(t)$ must be constant while v_J remains active. These conclusions follow from the learning rules for the bottom–up LTM traces z_{iJ}. Together, these properties imply that learning confirms the choice of v_J and does not trigger reset of v_J by the orienting subsystem.

Suppose that read-out of $V^{(J)}$ is first registered by F_1 at time $t = t_0$. By the $\frac{2}{3}$ Rule, $\mathbf{X}(t_0) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$. By (49), $z_{Ji}(t)$ begins to increase towards 1 if $i \in \mathbf{X}(t_0)$, and begins to decrease towards 0 if $i \notin \mathbf{X}(t_0)$. The Appendix shows that when v_J is active at F_2, each activity x_i in F_2 obeys the equation

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)(I_i + D_1 z_{Ji}) - (B_1 + C_1 x_i). \tag{89}$$

By (89), $x_i(t)$ increases if $z_{Ji}(t)$ increases, and $x_i(t)$ decreases if $z_{Ji}(t)$ decreases. Activities x_i which start out positive hereby become even larger, whereas activities x_i which start out non-positive become even smaller. In particular, $\mathbf{X}(t) = \mathbf{X}(t_0) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$ for all times $t \geq t_0$ at which v_J remains active.

We next prove that $T_J(t)$ increases, whereas all other $T_j(t)$ remain constant, while v_J is active. We suppose first that v_J is not an uncommitted node before considering the case in which v_J is an uncommitted node. While v_J remains active, the set $\mathbf{X}(t) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$. Thus

$$T_J(t) = D_2 \sum_{i \in \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)} z_{iJ}(t). \tag{90}$$

At time $t = t_0$, each LTM trace in (90) satisfies

$$z_{iJ}(t_0) \cong \frac{L}{L - 1 + |\mathbf{V}^{(J)}(t_0)|} \tag{91}$$

due to (79). While v_J remains active, each of these LTM traces responds to the fact that $\mathbf{X}(t) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$. By (47) and (52), each $z_{iJ}(t)$ with $i \in \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$ increases towards

$$\frac{L}{L - 1 + |\mathbf{I} \cap \mathbf{V}^{(J)}(t_0)|}, \tag{92}$$

each $z_{iJ}(t)$ with $i \notin \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$ decreases towards 0, and all other bottom–up

LTM traces $z_{iJ}(t)$ remain constant. A comparison of (91) with (92) shows that $T_J(t)$ in (90) can only increase while v_J remains active. In contrast, all other $T_j(t)$ are constant while v_J remains active.

If v_J is an uncommitted node, then no LTM trace $z_{iJ}(t)$ changes before time $t = t_0$. Thus

$$z_{iJ}(t_0) = z_{iJ}(0), \qquad i = 1, 2, \ldots, M. \tag{93}$$

By the Template Learning Inequality (75), $\mathbf{I} \cap \mathbf{V}^{(J)}(t_0) = \mathbf{I}$, so that (90) can be written as

$$T_J(t) = D_2 \sum_{i \in \mathbf{I}} z_{iJ}(t). \tag{94}$$

By (93) and the Direct Access Inequality (54),

$$z_{iJ}(t_0) < \frac{L}{L - 1 + M}, \qquad i = 1, 2, \ldots, M. \tag{95}$$

While v_J remains active, $\mathbf{X}(t) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0) = \mathbf{I}$, so that each $z_{iJ}(t)$ in (94) approaches the value

$$\frac{L}{L - 1 + |\mathbf{I}|}. \tag{96}$$

Since $|\mathbf{I}| \leq M$ for any input pattern I, a comparison of (95) and (96) shows that each $z_{iJ}(t)$ with $i \in \mathbf{I}$ increases while v_J remains active. In contrast, each $z_{iJ}(t)$ with $i \notin \mathbf{I}$ decreases towards zero and all other $z_{ij}(t)$ remain constant. Consequently, by (94), $T_J(t)$ increases and all other $T_j(t)$ are constant while v_J remains active. Thus learning confirms the choice of v_J. Hence the set $\mathbf{X}(t)$ remains constant and equal to $\mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$ while learning proceeds.

This last fact, along with the hypothesis that read-out of $V^{(J)}$ does not immediately cause reset of v_J, implies that learning cannot trigger reset of v_J. By the Reset Rule (73), the hypothesis that read-out of $V^{(J)}$ does not immediately cause reset of v_J implies that

$$\left|\mathbf{I} \cap \mathbf{V}^{(J)}(t_0)\right| = \left|\mathbf{X}(t_0)\right| \geq \rho|\mathbf{I}|. \tag{97}$$

The fact that $\mathbf{X}(t)$ does not change while v_J remains active implies that

$$\left|\mathbf{X}(t)\right| = \left|\mathbf{X}(t_0)\right| \geq \rho|\mathbf{I}| \tag{98}$$

and hence that learning does not trigger reset of v_J. Thus v_J remains active and learning in its LTM traces $z_{iJ}(t)$ and $z_{Ji}(t)$ can continue until the trial is ended. This completes the proof of Theorem 4.

Theorems 2–4 immediately imply the following important corollary, which illustrates how $\frac{2}{3}$ Rule matching, the learning laws, and the Reset Rule work together to prevent spurious reset events.

COROLLARY 1 (Reset by mismatch). *An active F_2 node v_J can be reset only by the orienting subsystem. Reset occurs when the template $V^{(J)}$ causes an F_1 mismatch such that*

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| < \rho|\mathbf{I}|. \tag{99}$$

Reset cannot be caused within the attentional subsystem due to reordering of adaptive filter signals T_j by template read-out or due to learning.

Theorem 4 implies another important corollary which characterizes how a template changes due to learning on a given trial.

COROLLARY 2 (Subset recoding). *If an F_2 node v_J is activated due to an input I and if read-out of $V^{(J)}$ at time $t = t_0$ implies that*

$$\left|\mathbf{I} \cap \mathbf{V}^{(J)}(t_0)\right| \geq \rho|\mathbf{I}|, \tag{100}$$

then v_J remains active until I shuts off, and the template set $\mathbf{V}^{(J)}(t)$ contracts from $\mathbf{V}^{(J)}(t_0)$ to $\mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$.

With these results in hand, we can now prove that the learning process self-stabilizes in response to an arbitrary list of binary input patterns.

THEOREM 5 (Stable category learning). *Assume the model hypotheses of Section 18. Then in response to an arbitrary list of binary input patterns, all* LTM *traces $z_{ij}(t)$ and $z_{ji}(t)$ approach limits after a finite number of learning trials. Each template set $\mathbf{V}^{(j)}$ remains constant except for at most $M - 1$ times $t_1^{(j)} < t_2^{(j)} < \cdots < t_{r_j}^{(j)}$ at which it progressively loses elements, leading to the*

Subset Recoding Property
$$\mathbf{V}^{(j)}\left(t_1^{(j)}\right) \supset \mathbf{V}^{(j)}\left(t_2^{(j)}\right) \supset \cdots \supset \mathbf{V}^{(j)}\left(t_{r_j}^{(j)}\right). \tag{101}$$

All LTM *traces oscillate at most once due to learning. The* LTM *traces $z_{ij}(t)$ and $z_{ji}(t)$ such that $i \notin V^{(j)}(t_1^{(j)})$ decrease monotonically to zero. The* LTM *traces $z_{ij}(t)$ and $z_{ji}(t)$ such that $i \in \mathbf{V}^{(j)}(t_{r_j}^{(j)})$ are monotone increasing functions. The* LTM *traces $z_{ij}(t)$ and $z_{ji}(t)$ such that $i \in \mathbf{V}^{(j)}(t_k^{(j)})$ but $i \notin \mathbf{V}^{(j)}(t_{k+1}^{(j)})$ can increase at times $t \leq t_{k+1}^{(j)}$ but can only decrease towards zero at times $t > t_{k+1}^{(j)}$.*

Proof. Suppose that an input pattern I is presented on a given trial and the Order Function satisfies

$$O_{j_1} > O_{j_2} > O_{j_3} > \cdots. \tag{76}$$

Then no learning occurs while F_2 nodes are searched in the order $v_{j_1}, v_{j_2}, \ldots$, by Theorem 3. If all F_2 nodes are reset by the search, then no learning occurs on that trial. If a node exists such that

$$|\mathbf{I} \cap \mathbf{V}^{(j)}| \geq \rho|\mathbf{I}|, \tag{102}$$

then search terminates at the first such node, v_{j_k}. Only the LTM traces z_{ij_k} and $z_{j_k i}$ can undergo learning on that trial, by Theorem 4. In particular, if an uncommitted

node v_{j_k} is reached by the search, then the Template Learning Inequality implies

$$|\mathbf{I} \cap \mathbf{V}^{(j_k)}| = |\mathbf{I} \cap \mathbf{V}^{(j_k)}(0)| = |\mathbf{I}| \geq \rho|\mathbf{I}| \tag{103}$$

so that its LTM traces undergo learning on that trial. In summary, learning on a given trial can change only the LTM traces of the F_2 node v_{j_k} at which the search ends.

Corollary 2 shows that the template set $\mathbf{V}^{(j_k)}$ of the node v_{j_k} is either constant or contracts due to learning. A contraction can occur on only a finite number of trials, because there are only finitely many nodes in F_1. In addition, there are only finitely many nodes in F_2, hence only finitely many template sets $\mathbf{V}^{(J)}$ can contract. The Subset Recoding Property is hereby proved.

The monotonicity properties of the LTM traces follow from the Subset Recoding Property and Theorem 4. Suppose for definiteness that the search on a given trial terminates at a node v_J in response to an input pattern I. Suppose moreover that the template set $\mathbf{V}^{(J)}(t)$ contracts from $\mathbf{V}^{(J)}(t_k^{(J)})$ to $\mathbf{V}^{(J)}(t_{k+1}^{(J)}) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_k^{(J)})$ due to read-out of the template $V^{(J)}(t_k^{(J)})$ on that trial. A comparison of (91) and (92) shows that each $z_{iJ}(t)$ with $i \in \mathbf{V}^{(J)}(t_{k+1}^{(J)})$ increases from

$$\frac{L}{L - 1 + |\mathbf{V}^{(J)}(t_k^{(J)})|} \tag{104}$$

to

$$\frac{L}{L - 1 + |\mathbf{V}^{(J)}(t_{k+1}^{(J)})|}, \tag{105}$$

that each $z_{iJ}(t)$ with $i \notin \mathbf{V}^{(J)}(t_{k+1}^{(J)})$ decreases towards zero, and that all other bottom–up LTM traces $z_{ij}(t)$ remain constant. In a similar fashion, each $z_{Ji}(t)$ with $i \in \mathbf{V}^{(J)}(t_{k+1}^{(J)})$ remains approximately equal to one, each $z_{Ji}(t)$ with $i \notin \mathbf{V}^{(J)}(t_{k+1}^{(J)})$ decreases towards zero, and all other top–down LTM traces $z_{ji}(t)$ remain constant.

Due to the Subset Recoding Property (101),

$$|\mathbf{V}^{(J)}(t_1^{(J)})| > |\mathbf{V}^{(J)}(t_2^{(J)})| > \cdots > |\mathbf{V}^{(J)}(t_{r_J}^{(J)})|. \tag{106}$$

Thus each LTM trace $z_{iJ}(t)$ with $i \in \mathbf{V}^{(J)}(t_{r_J}^{(J)})$ increases monotonically, as from (104) to (105), on the r_J trials where search ends at v_J and the template set $\mathbf{V}^{(J)}(t)$ contracts. On all other trials, these LTM traces remain constant. The other monotonicity properties are now also easily proved by combining the Subset Recoding Property (101) with the learning properties on a single trial. In particular, by the Subset Recoding Property, no LTM traces change after time

$$t = \max\{t_{r_j}^{(j)}: j = M + 1, M + 2, \ldots, N\}. \tag{107}$$

Thus all LTM traces approach their limits after a finite number of learning trials. This completes the proof of Theorem 5.

21. CRITICAL FEATURE PATTERNS AND PROTOTYPES

The property of stable category learning can be intuitively summarized using the following definitions.

The *critical feature pattern* at time t of a node v_j is the template $V^{(j)}(t)$. Theorem 5 shows that the critical feature pattern of each node v_j is progressively refined as the learning process discovers the set of features that can match all the input patterns which v_j codes. Theorem 5 also says that the network discovers a set of *self-stabilizing* critical feature patterns as learning proceeds. At any stage of learning, the set of all critical feature patterns determines the order in which previously coded nodes will be activated, via the Order Function

$$O_j = \frac{D_2 L |\mathbf{I} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|}. \tag{108}$$

The *Reset Function*

$$R_j = \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{I}|} \tag{109}$$

determines how many of these nodes will actually be searched, and thus which node may be recoded on each trial. In particular, an unfamiliar input pattern which has never before been experienced by the network will directly access a node v_{j_1} if the

Direct Access Conditions

$$O_{j_1} > \max(O_j: j \neq j_1) \quad \text{and} \quad R_{j_1} \geq \rho. \tag{110}$$

are satisfied.

An important example of direct access occurs when the input pattern I^* satisfies $\mathbf{I}^* = \mathbf{V}^{(j)}$, for some $j = M + 1, M + 2, \ldots, N$. Such an input pattern is called a *prototype*. Due to the Subset Recoding Property (101), at any given time a prototype pattern includes all the features common to the input patterns which have previously been coded by node v_j. Such a prototype pattern may never have been experienced itself. When an unfamiliar prototype pattern is presented for the first time, it will directly access its category v_j and is thus recognized. This property follows from Theorem 1, since v_j has perfectly learned I^*. Moreover, because $\mathbf{I}^* = \mathbf{V}^{(j)}$, a prototype is optimally matched by read-out of the template $V^{(j)}$.

A prototype generates an optimal match in the bottom–up filter, in the top–down template, and at F_1, even though it is unfamiliar. This is also true in human recognition data [20, 22, 23]. Theorem 5 thus implies that an ART system can discover, learn, and recognize stable prototypes of an arbitrary list of input patterns. An ART system also supports direct access by unfamiliar input patterns which are not prototypes, but which share invariant properties with learned prototypes, in the sense that they satisfy the Direct Access Conditions.

22. DIRECT ACCESS AFTER LEARNING SELF-STABILIZES

We can now prove that all patterns directly access their categories after the recognition learning process self-stabilizes. In order to discuss this property precisely, we define three types of learned templates with respect to an input pattern I:

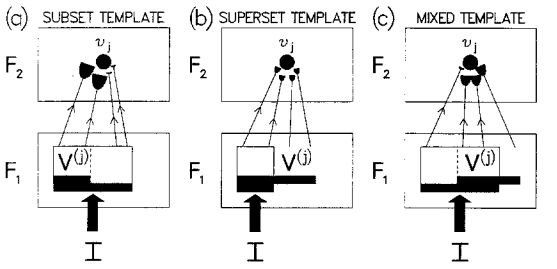

FIG. 11. Subset, superset, and mixed templates $V^{(j)}$ with respect to an input pattern I: In (a), (b), and (c), the lower black bar designates the set of F_1 nodes that receive positive bottom–up inputs due to I. The upper black bar designates the set of F_1 nodes that receive positive top–down inputs due to the template $V^{(j)}$. (a) denotes a subset template $V^{(j)}$ with respect to I. (b) denotes a superset template $V^{(j)}$ with respect to I. (c) denotes a mixed template $V^{(j)}$ with respect to I. When node v_j in F_2 is not an uncommitted node, the top–down LTM traces in the template $V^{(j)}$ are large if and only if the LTM traces in the corresponding bottom–up pathways are large (Sect. 15). The absolute bottom–up LTM trace size depends inversely upon the size $|\mathbf{V}^{(j)}|$ of $V^{(j)}$, due to the Weber Law Rule (Sect. 14). Larger LTM traces are drawn as larger endings on the bottom–up pathways. The arrow heads denote the pathways that are activated by I before any top–down template influences F_1.

subset templates, superset templates, and mixed templates. The LTM traces of a subset template V satisfy $\mathbf{V} \subseteq \mathbf{I}$: they are large only at a subset of the F_1 nodes which are activated by the input pattern I (Fig. 11a). The LTM traces of a superset template V satisfy $\mathbf{V} \supset \mathbf{I}$: they are large at all the F_1 nodes which are activated by the input pattern I, as well as at some F_1 nodes which are not activated by I (Fig. 11b). The LTM traces of a mixed template V are large at some, but not all, the F_1 nodes which are activated by the input pattern I, as well as at some F_1 nodes which are not activated by I: the set $\mathbf{I}$ is neither a subset nor a superset of $\mathbf{V}$ (Fig. 11c).

THEOREM 6 (Direct access after learning self-stabilizes). *Assume the model hypotheses of Section* 18. *After recognition learning has self-stabilized in response to an arbitrary list of binary input patterns, each input pattern I either has direct access to the node v_j which possesses the largest subset template with respect to I, or I cannot be coded by any node of F_2. In the latter case, F_2 contains no uncommitted nodes.*

Remark. The possibility that an input pattern cannot be coded by any node of F_2 is a consequence of the fact that an ART network self-stabilizes its learning in response to a list containing arbitrarily many input patterns no matter how many coding nodes exist in F_2. If a list contains many input patterns and F_2 contains only a few nodes, one does not expect F_2 to code all the inputs if the vigilance parameter ρ is close to 1.

Proof. Since learning has already stabilized, I can be coded only by a node v_j whose template $V^{(j)}$ is a subset template with respect to I. Otherwise, after template $V^{(j)} = V$ was read-out, the set $\mathbf{V}^{(j)}$ would contract from $\mathbf{V}$ to $\mathbf{I} \cap \mathbf{V}$ by Corollary 2 (Sect. 20), thereby contradicting the hypothesis that learning has already stabilized. In particular, input I cannot be coded by a node whose template is a superset template or a mixed template with respect to I. Nor can I be coded by an uncommitted node. Thus if I activates any node other than one with a subset template, that node must be reset by the orienting subsystem.

For the remainder of the proof, let v_J be the first F_2 node activated by I. We show that if $V^{(J)}$ is a subset template, then it is the subset template with the largest index set; and that if the orienting subsystem resets v_J, then it also resets all nodes with subset templates which get activated on that trial. Thus either the node with maximal subset template is directly accessed, or all nodes in F_2 that are activated by I are quickly reset by the orienting subsystem because learning has already self-stabilized.

If v_j is any node with a subset template $V^{(j)}$ with respect to I, then the Order Function

$$O_j = \frac{D_2 L|\mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|}, \tag{111}$$

by (108). Function O_j in (111) is an increasing function of $|\mathbf{V}^{(j)}|$. Thus if the first chosen node v_J has a subset template, then $V^{(J)}$ is the subset template with the largest index set.

If v_j is any node with a subset template $V^{(j)}$ with respect to I, then the Reset Function

$$R_j = \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{I}|} = \frac{|\mathbf{V}^{(j)}|}{|\mathbf{I}|}, \tag{112}$$

by (109). Once activated, such a node v_j will be reset if

$$R_j < \rho. \tag{113}$$

Thus if the node with the largest index set $\mathbf{V}^{(j)}$ is reset, (112) and (113) imply that all other nodes with subset templates will be reset.

Finally, suppose that v_J, the first node activated, does not have a subset template, but that some node v_j with a subset template is activated in the course of search. We need to show that $|\mathbf{I} \cap \mathbf{V}^{(j)}| = |\mathbf{V}^{(j)}| < \rho|\mathbf{I}|$, so that v_j is reset. Since v_j has a subset template,

$$O_j = \frac{D_2 L|\mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|}. \tag{111}$$

Since $|\mathbf{I} \cap \mathbf{V}^{(J)}| \leq |\mathbf{V}^{(J)}|$,

$$O_J = \frac{D_2 L|\mathbf{I} \cap \mathbf{V}^{(J)}|}{L-1+|\mathbf{V}^{(J)}|} \leq \frac{D_2 L|\mathbf{V}^{(J)}|}{L-1+|\mathbf{V}^{(J)}|}. \tag{114}$$

Since v_J was chosen first, $O_J > O_j$. Comparison of (111) and (114) thus implies that $|\mathbf{V}^{(J)}| > |\mathbf{V}^{(j)}|$. Using the properties $O_j < O_J$, $|\mathbf{I} \cap \mathbf{V}^{(J)}| < \rho|\mathbf{I}|$, and $|\mathbf{V}^{(J)}| > |\mathbf{V}^{(j)}|$ in turn, we find

$$\frac{|\mathbf{V}^{(j)}|}{L-1+|\mathbf{V}^{(j)}|} < \frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{L-1+|\mathbf{V}^{(J)}|} < \frac{\rho|\mathbf{I}|}{L-1+|\mathbf{V}^{(J)}|} < \frac{\rho|\mathbf{I}|}{L-1+|\mathbf{V}^{(j)}|}, \tag{115}$$

which implies that

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| = |\mathbf{V}^{(J)}| < \rho|\mathbf{I}|. \tag{116}$$

Therefore all F_2 nodes are reset if v_J is reset. This completes the proof of Theorem 6.

Theorem 6 shows that, in response to any familiar input pattern I, the network knows how to directly access the node v_j whose template $V^{(j)}$ corresponds to the prototype $I^* = V^{(j)}$ which is closest to I among all prototypes learned by the network. Because direct access obviates the need for search, recognition of familiar input patterns and of unfamiliar patterns that share categorical invariants with familiar patterns is very rapid no matter how large or complex the learned recognition code may have become. Grossberg and Stone [12] have, moreover, shown that the variations in reaction times and error rates which occur during direct access due to prior priming events are consistent with data collected from human subjects in lexical decision experiments and word familiarity and recall experiments.

Theorems 5 and 6 do not specify how many list presentations and F_2 nodes are needed to learn and recognize an arbitrary list through direct access. We make the following conjecture: in the fast learning case, if F_2 has at least n nodes, then each member of a list of n input patterns which is presented cyclically will have direct access to an F_2 node after at most n list presentations.

Given arbitrary lists of input patterns, this is the best possible result. If the vigilance parameter ρ is close to 1 and if a nested set of n binary patterns is presented in order of decreasing size, then exactly n list presentations are required for the final code to be learned. On the other hand, if a nested set of n patterns is presented in order of increasing size, then only one list presentation is required for the final code to be learned. Thus the number of trials needed to stabilize learning in the fast learning case depends upon both the ordering and the internal structure of the input patterns, as well as upon the vigilance level.

23. ORDER OF SEARCH: MATHEMATICAL ANALYSIS

The Order Function

$$O_j = \frac{D_2 L|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|} \tag{108}$$

for previously coded nodes v_j shows that search order is determined by two opposing tendencies. A node v_j will be searched early if $|\mathbf{I} \cap \mathbf{V}^{(j)}|$ is large and if $|\mathbf{V}^{(j)}|$ is small. Term $|\mathbf{I} \cap \mathbf{V}^{(j)}|$ is maximized if $V^{(j)}$ is a superset template of I. Term $|\mathbf{V}^{(j)}|$ is small if $V^{(j)}$ codes only a few features. The relative importance of the template intersection $|\mathbf{I} \cap \mathbf{V}^{(j)}|$ and the template size $|\mathbf{V}^{(j)}|$ is determined by the size of $L - 1$ in (108). If $L - 1$ is small, both factors are important. If $L - 1$ is large, the template intersection term dominates search order. The next theorem completely characterizes the search order in the case that $L - 1$ is small.

THEOREM 7 (Search order). *Assume the model hypotheses of Section* 18. *Suppose that input pattern I satisfies*

$$L - 1 \leq \frac{1}{|\mathbf{I}|} \tag{117}$$

and

$$|\mathbf{I}| \leq M - 1. \tag{118}$$

Then F_2 nodes are searched in the following order, if they are reached at all.

Subset templates with respect to I are searched first, in order of decreasing size. If the largest subset template is reset, then all subset templates are reset. If all subset templates have been reset and if no other learned templates exist, then the first uncommitted node to be activated will code I. If all subset templates are searched and if there exist learned superset templates but no mixed templates, then the node with the smallest superset template will be activated next and will code I. If all subset templates are searched and if both superset templates $V^{(J)}$ and mixed templates $V^{(j)}$ exist, then v_j will be searched before v_J if and only if

$$|\mathbf{V}^{(j)}| < |\mathbf{V}^{(J)}| \quad \text{and} \quad \frac{|\mathbf{I}|}{|\mathbf{V}^{(J)}|} < \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{V}^{(j)}|}. \tag{119}$$

If all subset templates are searched and if there exist mixed templates but no superset templates, then a node v_j with a mixed template will be searched before an uncommitted node v_J if and only if

$$\frac{L|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|} > \sum_{i \in \mathbf{I}} z_{iJ}(0). \tag{120}$$

The proof is based upon the following lemma.

LEMMA 1. *If* (117) *holds, then for any pair of F_2 nodes v_J and v_j with learned templates, $O_J > O_j$ if either*

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{V}^{(J)}|} > \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{V}^{(j)}|} \tag{121}$$

or

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{V}^{(J)}|} = \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{V}^{(j)}|} \quad \text{and} \quad |\mathbf{V}^{(J)}| > |\mathbf{V}^{(j)}|. \tag{122}$$

Proof of Lemma 1. We need to show that if either (121) or (122) holds, then $O_J > O_j$. By (108), $O_J > O_j$ is equivalent to

$$\begin{aligned} &|\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}| \cdot |\mathbf{V}^{(J)}| \\ &\quad + (L - 1)\big[|\mathbf{I} \cap \mathbf{V}^{(J)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}|\big] > 0. \end{aligned} \tag{123}$$

Suppose that (121) holds. Then:

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}| \cdot |\mathbf{V}^{(J)}| > 0. \tag{124}$$

Since $L > 1$, inequality (123) then follows at once if $[|\mathbf{I} \cap \mathbf{V}^{(J)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}|] \geq 0$.

Suppose that $|\mathbf{I} \cap \mathbf{V}^{(j)}| > |\mathbf{I} \cap \mathbf{V}^{(J)}|$. Each term in (124) is an integer. The entire left-hand side of (124) is consequently a positive integer, so

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}| \cdot |\mathbf{V}^{(J)}| \geq 1 > \frac{|\mathbf{I}| - 1}{|\mathbf{I}|}. \tag{125}$$

Inequality (124) also implies that $|\mathbf{I} \cap \mathbf{V}^{(J)}| \geq 1$, and in general $|\mathbf{I}| \geq |\mathbf{I} \cap \mathbf{V}^{(j)}|$. Thus by (117) and (125),

$$\begin{aligned} |\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}| \cdot |\mathbf{V}^{(J)}| &> (L-1)(|\mathbf{I}| - 1) \\ &\geq (L-1)\big[|\mathbf{I} \cap \mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(J)}|\big]. \end{aligned} \tag{126}$$

Inequality (126) implies (123), and hence $O_J > O_j$.

Suppose next that (122) holds. Then

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(j)}| - |\mathbf{I} \cap \mathbf{V}^{(j)}| \cdot |\mathbf{V}^{(J)}| = 0. \tag{127}$$

Also, $|\mathbf{V}^{(J)}| > |\mathbf{V}^{(j)}|$, so

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{I} \cap \mathbf{V}^{(j)}|} = \frac{|\mathbf{V}^{(J)}|}{|\mathbf{V}^{(j)}|} > 1. \tag{128}$$

Equations (127) and (128) imply (123), thereby completing the proof of Lemma 1.

We can now prove the theorem.

Proof of Theorem 7. First we show that a node v_J with a subset template is searched before any node v_j with a mixed or superset template. Since $\mathbf{I} \cap \mathbf{V}^{(J)} = \mathbf{V}^{(J)}$ but $\mathbf{I} \cap \mathbf{V}^{(j)}$ is a proper subset of $\mathbf{V}^{(j)}$,

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{V}^{(J)}|} = \frac{|\mathbf{V}^{(J)}|}{|\mathbf{V}^{(J)}|} = 1 > \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{V}^{(j)}|}. \tag{129}$$

By (121) in Lemma 1, $O_J > O_j$. Thus all subset templates are searched before mixed templates or learned superset templates.

We next show that a node v_J with a subset template is also searched before any uncommitted node v_j. Since

$$O_j = D_2 \sum_{i \in \mathbf{I}} z_{ij}, \tag{130}$$

the Direct Access Inequality (54) implies that

$$O_j < \frac{D_2 L |\mathbf{I}|}{L - 1 + M}. \tag{131}$$

The right-hand side of (131) is an increasing function of L. Thus by (117).

$$\frac{D_2 L |\mathbf{I}|}{L - 1 + M} \leq \frac{D_2 (|\mathbf{I}|^{-1} + 1)|\mathbf{I}|}{|\mathbf{I}|^{-1} + M} = \frac{D_2 (1 + |\mathbf{I}|)}{|\mathbf{I}|^{-1} + M}. \tag{132}$$

Inequality (118) implies that

$$\frac{D_2(1 + |\mathbf{I}|)}{|\mathbf{I}|^{-1} + M} \leq \frac{D_2 M}{|\mathbf{I}|^{-1} + M} < D_2. \tag{133}$$

On the other hand, since $|\mathbf{V}^{(J)}| \geq 1$,

$$O_J = \frac{D_2 L|\mathbf{V}^{(J)}|}{L - 1 + |\mathbf{V}^{(J)}|} \geq \frac{D_2 L \cdot 1}{L - 1 + 1} = D_2. \tag{134}$$

Inequalities (131)–(134) together imply $O_J > O_j$.

If v_J has a subset template, then $|\mathbf{I} \cap \mathbf{V}^{(J)}| = |\mathbf{V}^{(J)}|$. Thus all nodes with subset templates have the same ratio $|\mathbf{I} \cap \mathbf{V}^{(J)}|\,|\mathbf{V}^{(J)}|^{-1} = 1$. By (122) in Lemma 1, nodes with subset templates are searched in the order of decreasing template size.

If all subset templates are searched and if no other learned templates exist, then an uncommitted node will be activated. This node codes I because it possesses an unlearned superset template that does not lead to F_2 reset.

Suppose all subset templates have been searched and that there exist learned superset templates but no mixed templates. If node v_J has a superset template $V^{(J)}$, then

$$O_J = \frac{D_2 L|\mathbf{I}|}{L - 1 + |\mathbf{V}^{(J)}|}. \tag{135}$$

By (135), the first superset node to be activated is the node v_J whose template is smallest. Node v_J is chosen before any uncommitted node v_j because, by (54),

$$O_J \geq \frac{D_2 L|\mathbf{I}|}{L - 1 + M} > D_2 \sum_{i \in \mathbf{I}} z_{ij}(0) = O_j. \tag{136}$$

If v_J is activated, it codes I because its template satisfies

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| = |\mathbf{I}| \geq \rho|\mathbf{I}|. \tag{137}$$

Suppose that all subset templates are searched and that a superset template $V^{(J)}$ and a mixed template $V^{(j)}$ exist. We prove that $O_j > O_J$ if and only if (119) holds. Suppose that (119) holds. Then also

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{V}^{(J)}|} = \frac{|\mathbf{I}|}{|\mathbf{V}^{(J)}|} < \frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{|\mathbf{V}^{(j)}|}. \tag{138}$$

By condition (121) of Lemma 1, $O_j > O_J$. Conversely, suppose that $O_j > O_J$. Then

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(j)}|}{L - 1 + |\mathbf{V}^{(j)}|} > \frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{L - 1 + |\mathbf{V}^{(J)}|} = \frac{|\mathbf{I}|}{L - 1 + |\mathbf{V}^{(J)}|}. \tag{139}$$

Since $V^{(J)}$ is a mixed template with respect to I, $|\mathbf{I} \cap \mathbf{V}^{(J)}| < |\mathbf{I}|$. Thus (139) implies that $|\mathbf{V}^{(J)}| < |\mathbf{V}^{(J)}|$ as well as

$$|\mathbf{I} \cap \mathbf{V}^{(J)}| \cdot |\mathbf{V}^{(J)}| - |\mathbf{I}| \cdot |\mathbf{V}^{(J)}| > (L-1)\left[|\mathbf{I}| - |\mathbf{I} \cap \mathbf{V}^{(J)}|\right] > 0, \quad (140)$$

from which (119) follows. This completes the proof of Theorem 7.

Note that Lemma 1 also specifies the order of search among mixed templates. If all the activated mixed template nodes are reset, then the node v_J with the minimal superset template will code I. Unless (120) holds, it is possible for an uncommitted node v_J to code I before a node with a mixed template v_j is activated. Inequality (120) does not automatically follow from the Direct Access Inequality (54) because $|\mathbf{I} \cap \mathbf{V}^{(J)}|$ may be much smaller than $|\mathbf{I}|$ when $V^{(J)}$ is a mixed template.

24. ORDER OF SEARCH: COMPUTER SIMULATIONS

Figures 12 and 13 depict coding sequences that illustrate the order of search specified by Theorem 7 when $(L-1)$ is small and when the vigilance parameter ρ is close to 1. In Fig. 12, each of nine input patterns was presented once. Consider the order of search that occurred in response to the final input pattern I that was presented on trial 9. By trial 8, nodes v_{M+1} and v_{M+2} had already encoded subset templates of this input pattern. On trial 9, these nodes were therefore searched in order of decreasing template size. Nodes v_{M+3}, v_{M+4}, v_{M+5}, and v_{M+6} had encoded mixed templates of the input pattern. These nodes were searched in the order $v_{M+3} \rightarrow v_{M+5} \rightarrow v_{M+4}$. This search order was not determined by template size *per se*, but was rather governed by the ratio $|\mathbf{I} \cap \mathbf{V}^{(J)}|\,|\mathbf{V}^{(J)}|^{-1}$ in (121) and (122). These ratios for nodes v_{M+3}, v_{M+5}, and v_{M+4} were $\frac{9}{10}$, $\frac{14}{16}$, and $\frac{7}{8}$, respectively. Since $\frac{14}{16} = \frac{7}{8}$, node v_{M+5} was searched before node v_{M+4} because $|\mathbf{V}^{(M+5)}| = 16 > 8 =$

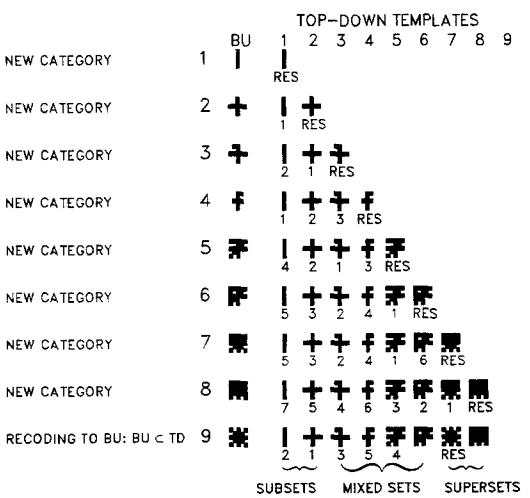

FIG. 12. Computer simulation to illustrate order of search: On trial 9, the system first searches subset templates, next searches some, but not all, mixed templates, and finally recodes the smallest superset template. A smaller choice of vigilance parameter could have terminated the search at a subset template or mixed template node.

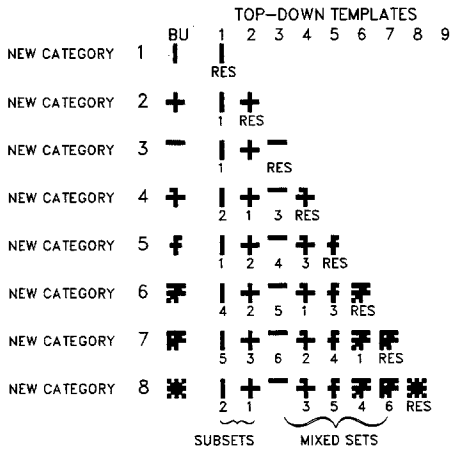

FIG. 13. Computer simulation to illustrate order of search: Unlike the search described in Fig. 12, no learned superset template exists when the search begins on trial 8. Consequently, the system first searches subset templates, next searches mixed templates, and finally terminates the search by coding a previously uncommitted node.

$|\mathbf{V}^{(M+4)}|$. The mixed template node v_{M+6} was not searched. After searching v_{M+5}, the network activated the node v_{M+7} which possessed the smallest superset template. A comparison of rows 8 and 9 in column 7 shows how the superset template of v_{M+7} was recoded to match the input pattern. By (119), the superset template node v_{M+7} was searched before the mixed template node v_{M+6} because the ratio $|\mathbf{I}|\,|\mathbf{V}^{(M+7)}|^{-1} = \frac{17}{21}$ was larger than $|\mathbf{I} \cap \mathbf{V}^{(M+6)}|\,|\mathbf{V}^{(M+6)}|^{-1} = \frac{14}{18}$.

The eight input patterns of Fig. 13 illustrate a search followed by coding of an uncommitted node. The last input pattern I in Fig. 13 is the same as the last input pattern in Fig. 12. In Fig. 13, however, there are no superset templates corresponding to input pattern I. Consequently I was coded by a previously uncommitted node v_{M+8} on trial 8. On trial 8 the network searched nodes with subset templates in the order $v_{M+2} \to v_{M+1}$ and the mixed template nodes in the order $v_{M+4} \to v_{M+6} \to v_{M+5} \to v_{M+7}$. The mixed template node v_{M+3} was not searched because its template badly mismatched the input pattern I and thus did not satisfy (120). Instead, the uncommitted node v_{M+8} was activated and learned a template that matched the input pattern. If $(L - 1)$ is not small enough to satisfy inequality (117), then mixed templates or superset templates may be searched before subset templates. For all $L > 1$, however, Theorem 6 implies that all input patterns have direct access to their coding nodes after the learning process equilibrates.

25. BIASING THE NETWORK TOWARDS UNCOMMITTED NODES

Another effect of choosing L large is to bias the network to choose uncommitted nodes in response to unfamiliar input patterns I. To understand this effect, suppose that for all i and j,

$$z_{ij}(0) \cong \frac{L}{L - 1 + M}; \tag{71}$$

Then when I is presented, an uncommitted node is chosen before a coded node v_j if

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{L - 1 + |\mathbf{V}^{(J)}|} < \frac{|\mathbf{I}|}{L - 1 + M}. \tag{141}$$

This inequality is equivalent to

$$\frac{|\mathbf{I} \cap \mathbf{V}^{(J)}|}{|\mathbf{I}|} < \frac{L - 1 + |\mathbf{V}^{(J)}|}{L - 1 + M}. \tag{142}$$

As L increases, the ratio

$$\frac{L - 1 + |\mathbf{V}^{(J)}|}{L - 1 + M} \to 1, \tag{143}$$

whereas the left-hand side of (142) is always less than or equal to 1. Thus for large values of L, the network tends to code unfamiliar input patterns into new categories, even if the vigilance parameter ρ is small. As L increases, the automatic scaling property (Sect. 2A) of the network also becomes weaker, as does the tendency to search subset templates first.

Recall that parameter L describes the relative strength of the bottom–up competition among LTM traces which gives rise to the Weber Law Rule (Sect. 12B), with smaller L corresponding to stronger LTM competition. Thus the structural process of LTM competition works with the state-dependent process of attentional vigilance to control how coarse the learned categories will be.

26. COMPUTER SIMULATION OF SELF-SCALING COMPUTATIONAL UNITS: WEIGHING THE EVIDENCE

We can now understand quantitatively how the network automatically rescales its matching and signal-to-noise criteria in the computer simulations of Fig. 7. On the first four presentations, the input patterns are presented in the order *ABAB*. By trial 2, learning is complete. Pattern A directly accesses node v_{M+1} on trial 3, and pattern B directly accesses node v_{M+2} on trial 4. Thus patterns A and B are coded within different categories. On trials 5–8, patterns C and D are presented in the order *CDCD*. Patterns C and D are constructed from patterns A and B, respectively, by adding identical upper halves to A and B. Thus, pattern C differs from pattern D at the same locations where pattern A differs from pattern B. However, because patterns C and D represent many more active features than patterns A and B, the difference between C and D is treated as noise and is deleted from the critical feature pattern of v_{M+3} which codes both C and D, whereas the difference between A and B is considered significant and is included within the critical feature patterns of v_{M+1} and v_{M+2}.

The core issue in the network's different categorization of patterns A and B versus patterns C and D is the following: Why on trial 2 does B reject the node v_{M+1} which has coded A, whereas D on trial 6 accepts the node v_{M+3} which has coded C? This occurs despite the fact that the mismatch between B and $V^{(M+1)}$ equals the mismatch between D and $V^{(M+3)}$:

$$|\mathbf{B}| - |\mathbf{B} \cap \mathbf{V}^{(M+1)}| = 3 = |\mathbf{D}| - |\mathbf{D} \cap \mathbf{V}^{(M+3)}|. \tag{144}$$

The reason is that

$$\frac{|\mathbf{B} \cap \mathbf{V}^{(M+1)}|}{|\mathbf{B}|} = \frac{8}{11}, \tag{145}$$

whereas

$$\frac{|\mathbf{D} \cap \mathbf{V}^{(M+3)}|}{|\mathbf{D}|} = \frac{14}{17}. \tag{146}$$

In this simulation, the vigilance parameter $\rho = 0.8$. Thus

$$\frac{|\mathbf{B} \cap \mathbf{V}^{(M+1)}|}{|\mathbf{B}|} < \rho < \frac{|\mathbf{D} \cap \mathbf{V}^{(M+3)}|}{|\mathbf{D}|}. \tag{147}$$

By (73), pattern B resets v_{M+1} on trial 2 but D does not reset v_{M+3} on trial 6. Consequently, B is coded by a different category than A, whereas D is coded by the same category as C.

27. CONCLUDING REMARKS: SELF-STABILIZATION AND UNITIZATION WITHIN ASSOCIATIVE NETWORKS

Two main conclusions of our work are especially salient. First, the code learning process is one of progressive refinement of distinctions. The distinctions that emerge are the resultant of all the input patterns which the network ever experiences, rather than of some preassigned features. Second, the matching process compares whole patterns, not just separate features. It may happen that two different input patterns to F_1 overlap a template at the same set of feature detectors, yet the network will reset the F_2 node in response to one input but not the other. The degree of mismatch of template pattern and input pattern *as a whole* determines whether coding or reset will occur. Thus the learning of categorical invariants resolves two opposing tendencies. As categories grow larger, and hence code increasingly global invariants, the templates which define them become smaller, as they discover and base the code on sets of critical feature patterns, or prototypes, rather than upon familiar pattern exemplars. This article shows how these two opposing tendencies can be resolved within a self-organizing system, leading to dynamic equilibration, or self-stabilization, of recognition categories in response to an arbitrary list of arbitrarily many binary input patterns. This self-stabilization property is of major importance for the further development of associative networks and the analysis of cognitive recognition processes.

Now that properties of self-organization, self-stabilization, and self-scaling are completely understood within the class of ART networks described herein, a number of generalizations also need to be studied. Within this article, an input pattern to level F_1 is globally grouped at F_2 when the F_2 population which receives the maximal input from the $F_1 \rightarrow F_2$ adaptive filter is chosen for short term memory (STM) storage. Within the total architecture of an ART system, even this simple type of F_2 reaction to the $F_1 \rightarrow F_2$ adaptive filter leads to powerful coding properties. On the other hand, a level F_2 which makes global choices must be viewed as a special case of a more general design for F_2.

If the second processing stage F_2 makes a choice, then later processing stages which are activated by F_2 alone could not further analyse the input pattern across F_1. The coding hierarchy for individual input patterns would end at the choice, or global grouping, stage. By contrast, a coding scheme wherein F_2 generates a spatially distributed representation of the F_1 activity pattern, rather than a choice, could support subsequent levels $F_3, F_4, \ldots, F_n$ for coding multiple groupings, or chunks, and thus more abstract invariants of an input pattern. This possibility raises many issues concerning the properties of these configurations and their invariants, and of the architectural constraints which enable a multilevel coding hierarchy to learn and recognize distributed invariants in a stable and globally self-consistent fashion.

A parallel neural architecture, called a *masking field* [9, 11, 12, 24–26, 36] is a type of circuit design from which F_2—and by extension higher levels $F_3, F_4, \ldots, F_n$—may be fashioned to generate distributed representations of filtered input patterns. Masking field properties are of value for visual object recognition, speech recognition, and higher cognitive processes. Indeed, the same circuit design can be used for the development of general spatially distributed self-organizing recognition codes. The purpose of a masking field is to detect simultaneously, and weight properly in STM, all salient parts, or groupings, of an input pattern. The pattern as a whole is but one such grouping. A masking field generates a spatially distributed, yet unitized, representation of the input pattern in STM. Computer simulations of how a masking field can detect and learn unitized distributed representations of an input are found in Cohen and Grossberg [24–26]. Much further work needs to be done to understand the design of ART systems all of whose levels F_i are masking fields.

Other useful generalizations of the ART system analysed herein include systems whose learning rate is slow relative to the time scale of a single trial; systems in which forgetting of LTM values can occur; systems which process continuous as well as binary input and output patterns; and systems in which Weber Law processing is realized through competitive STM interactions among F_1 nodes rather than competitive LTM interactions among bottom–up LTM traces (Sect. 12B). All of these generalizations will be considered in our future articles of this series.

Preprocessing of the input patterns to an ART system is no less important than choosing levels F_i capable of supporting a hierarchy of unitized codes of parts and wholes. In applications to visual object recognition, neural circuits which generate pre-attentively completed segmentations of a visual image before these completed segmentations generate inputs to an ART network have recently been constructed [37–40]. In applications to adaptive speech recognition, inputs are encoded as STM patterns of temporal order information across item representations before these STM patterns generate inputs to an ART network [9, 11–13, 24, 26, 36]. Further work needs to be done to characterize these preprocessing stages and how they are joined to their ART coding networks. Although a great deal of work remains to be done, results such as those in the present article amply illustrate that the whole is much greater than the sum of its parts both in human experience and in self-organizing models thereof.

APPENDIX

Table 1 lists the constraints on the dimensionless model parameters for the system summarized in Section 18. We will now show that the $\frac{2}{3}$ Rule holds when these

constraints are satisfied. Then we describe four alternative, but dynamically equivalent, systems for realizing the $\frac{2}{3}$ Rule and attentional gain control.

Recall that x_i $(i = 1 \ldots M)$ denotes the STM activity of an F_1 node v_i; that x_j $(j = M + 1 \ldots N)$ denotes the STM activity of an F_2 node v_j; that z_{ij} denotes the strength of the LTM trace in the bottom–up pathway from v_i to v_j; that z_{ji} denotes the strength of the LTM trace in the top–down pathway from v_j to v_i; that I_i denotes the bottom–up input to v_i; that $\mathbf{I}$ denotes the set of indices $i \in \{1 \ldots M\}$ v_i such that $I_i > 0$; that $\mathbf{X} = \mathbf{X}(t)$ denotes the set of indices i such that $x_i(t) > 0$; and that $\mathbf{V}^{(J)} = \mathbf{V}^{(J)}(t)$ denotes the set of indices i such that $z_{Ji}(t) > \bar{z}$.

Combining equations (8), (10), (11), and (12), we find the following equation for the ith STM trace of F_1

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)\left(I_i + D_1 \sum_j f(x_j) z_{ji}\right) - (B_1 + C_1 x_i) \sum_j f(x_j). \quad \text{(A1)}$$

When F_2 is inactive, all top–down signals $f(x_j) = 0$. Hence by (A1),

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i) I_i. \quad \text{(A2)}$$

When the F_2 node v_J is active, only the top–down signal $f(x_J)$ is nonzero. Since $f(x_J) = 1$,

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)(I_i + D_1 z_{Ji}) - (B_1 + C_1 x_i). \quad \text{(A3)}$$

Since each x_i variable changes rapidly relative to the rate of change of the LTM trace z_{Ji} (since $0 < \varepsilon \ll 1$), then x_i is always close to its steady state, $dx_i/dt = 0$. By (A2), then

$$x_i \cong \frac{I_i}{1 + A_1 I_i} \qquad \text{if } F_2 \text{ is inactive} \quad \text{(A4)}$$

and, by (A3),

$$x_i \cong \frac{I_i + D_1 z_{Ji} - B_1}{1 + A_1(I_i + D_1 z_{Ji}) + C_1} \qquad \text{if the } F_2 \text{ node } v_J \text{ is active.} \quad \text{(A5)}$$

The $\frac{2}{3}$ Rule, as defined by

$$\mathbf{X} = \begin{cases} \mathbf{I} & \text{if } F_2 \text{ is inactive} \\ \mathbf{I} \cap \mathbf{V}^{(J)} & \text{if the } F_2 \text{ node } v_J \text{ is active,} \end{cases} \quad (47)$$

can be derived as follows. Note first that (A4) implies that, when F_2 is inactive, $x_i > 0$ iff $I_i > 0$; i.e., $\mathbf{X} = \mathbf{I}$. On the other hand, if v_J is active, (A5) implies that

$$x_i > 0 \qquad \text{iff } z_{Ji} > \frac{B_1 - I_i}{D_1}. \quad \text{(A6)}$$

The $\frac{2}{3}$ Rule requires that x_i be positive when the F_1 node v_i is receiving large inputs, both top–down and bottom–up. Thus setting $z_{Ji} = 1$ And $I_i = 1$ (their maximal values) in (A6) implies the constraint:

$$1 > \frac{B_1 - 1}{D_1}. \tag{A7}$$

The $\frac{2}{3}$ Rule also requires that x_i be negative if v_i receives no top–down input, even if the bottom–up input is large. Thus setting $z_{Ji} = 0$ and $I_i = 1$ in (A6) implies the constraint:

$$0 < \frac{B_1 - 1}{D_1}. \tag{A8}$$

Finally, the $\frac{2}{3}$ Rule requires that x_i be negative if v_i receives no bottom–up input, even if the top–down input is large. Thus setting $I_i = 0$ and $z_{Ji} = 1$ in (A6) implies the constraint

$$1 < \frac{B_1}{D_1}. \tag{A9}$$

Inequalities (A7), (A8), and (A9) are summarized by the

$\frac{2}{3}$ Rule Inequalities

$$\max\{1, D_1\} < B_1 < 1 + D_1. \tag{68}$$

Since $0 \leq I_i \leq 1$, (A6) also shows that if v_J is active and if

$$z_{Ji}(t) \leq \frac{B_1 - 1}{D_1}, \tag{A10}$$

then $x_i(t) \leq 0$; i.e., $i \notin \mathbf{X}$. However if $i \notin \mathbf{X}$, z_{Ji} decays toward 0 whenever v_J is active. Thus if (A10) is true at some time $t = t_0$, it remains true for all $t \geq t_0$. Therefore

$$\bar{z} \equiv \frac{B_1 - 1}{D_1} \tag{69}$$

is the critical top–down LTM strength such that if $z_{Ji}(t_0) \leq \bar{z}$, then $z_{Ji}(t) \leq \bar{z}$ for all $t \geq t_0$. Whenever v_J is active and $t \geq t_0$, the F_1 node v_i will be inactive.

Figure 14 depicts four ways in which attentional gain control can distinguish bottom–up and top–down processing to implement the $\frac{2}{3}$ Rule. All of these systems generate the same asymptote (A5) when F_2 is active, and the same asymptotes, up to a minor change in parameters, when F_2 is inactive. The parameters in all four systems are defined to satisfy the constraints in Table 1.

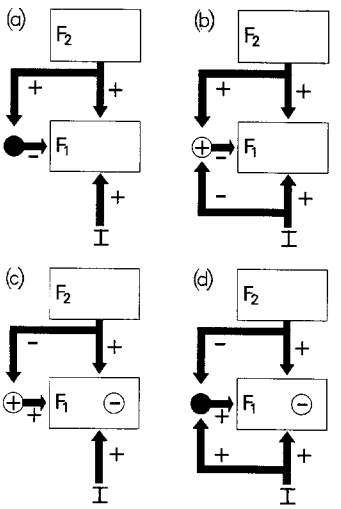

FIG. 14. Design variations for realizing $\frac{2}{3}$ Rule matching properties at F_1: In (a) and (b), F_2 excites the gain control channel, whereas in (c) and (d), F_2 inhibits the gain control channel. In (b), the input pattern I inhibits the gain control channel, whereas in (d), I excites the gain control channel. In (a) and (d), the gain control channel phasically reacts to its inputs (closed circles). Activation of the gain control channel in (a) nonspecifically inhibits F_1, and in (d) nonspecifically excites F_1. In (b) and (c), the gain control channel is tonically, or persistently, active in the absence of inputs (open circles surrounding plus signs). Activation of the gain control channel in (b) nonspecifically inhibits F_1, and in (c) nonspecifically excites F_1. In (c) and (d), the F_1 cells are maintained in a state of tonic hyperpolarization, or inhibition, in the absence of external inputs (open circles surrounding minus signs). All four cases lead to equivalent dynamics.

In Fig. 14a, F_2 can phasically excite the gain control channel, which thereupon nonspecifically inhibits the cells of F_1. Thus

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)\left(I_i + D_1 \sum_j f(x_j) z_{ji}\right) - (B_1 + C_1 x_i) G_1, \quad \text{(A11)}$$

where

$$G_1 = \begin{cases} 0 & \text{if } I \text{ is active and } F_2 \text{ is inactive} \\ 1 & \text{if } I \text{ is inactive and } F_2 \text{ is active} \\ 1 & \text{if } I \text{ is active and } F_2 \text{ is active} \\ 0 & \text{if } I \text{ is inactive and } F_2 \text{ is inactive.} \end{cases} \quad \text{(A12)}$$

In other words $G_1 = \sum_j f(x_j)$. Thus (A11) is just (A1) in a slightly different notation.

In Fig. 14b, the plus sign within an open circle in the gain control channel designates that the gain control cells, in the absence of any bottom–up or top–down signals, are endogenously maintained at an equilibrium potential which exceeds their output threshold. Output signals from the gain control cells nonspecifically inhibit the cells of F_1. In short, the gain control channel tonically, or persistently, inhibits F_1 cells in the absence of bottom–up or top–down signals. Bottom–up and

top–down signals phasically modulate the level of nonspecific inhibition. In particular, a bottom–up input alone totally inhibits the gain control channel, thereby disinhibiting the cells of F_1. A top–down signal alone maintains the inhibition from the gain control channel, because the inhibition is either on or off, and is thus not further increased by F_2. When both a bottom–up input and a top–down signal are active, their inputs to the gain control channel cancel, thereby again maintaining the same level of inhibition to F_1. The STM equations at F_1 are

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)\left(I_i + D_1 \sum_j f(x_j) z_{ji}\right) - (B_1 + C_1 x_i) G_2. \quad \text{(A13)}$$

where

$$G_2 = \begin{cases} 0 & \text{if } I \text{ is active and } F_2 \text{ is inactive} \\ 1 & \text{if } I \text{ is inactive and } F_2 \text{ is active} \\ 1 & \text{if } I \text{ is active and } F_2 \text{ is active} \\ 1 & \text{if } I \text{ is inactive and } F_2 \text{ is inactive.} \end{cases} \quad \text{(A14)}$$

The equilibrium activities of x_i are as follows. If I is active and F_2 is inactive, then (A4) again holds. If I is inactive and F_2 is active, then (A5) again holds. Equation (A5) also holds if I is active and F_2 is active. If I is inactive and F_2 is inactive, then

$$x_i \cong \frac{-B_1}{1 + C_1}, \quad \text{(A15)}$$

which is negative; hence no output signals are generated.

In Fig. 14c, as in Fig. 14b, the gain control cells are tonically active (plus sign in open circle). In Fig. 14c, however, these cells nonspecifically excite the cells of F_1. In the absence of any external signals, F_1 cells are maintained in a state of tonic hyperpolarization, or negative activity (denoted by the minus sign in the open circle). The tonic excitation from the gain control cells balances the tonic inhibition due to hyperpolarization and thereby maintains the activity of F_2 cells near their output threshold of zero. A bottom–up input can thereby excite F_1 cells enough for them to generate output signals. When top–down signals are active, they inhibit the gain control cells. Consequently those F_1 cells which do not receive bottom–up or top–down signals become hyperpolarized. Due to tonic hyperpolarization, F_1 cells which receive a bottom–up signal or a top–down signal, but not both, cannot exeed their output threshold. Only F_1 cells at which large top–down and bottom–up signals converge can generate an output signal.

The STM equations at F_1 are

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)\left(I_i + D_1 \sum_j f(x_j) z_{ji} + B_1 G_3\right) - (B_1 + C_1 x_i), \quad \text{(A16)}$$

where

$$G_3 = \begin{cases} 1 & \text{if } I \text{ is active and } F_2 \text{ is inactive} \\ 0 & \text{if } I \text{ is inactive and } F_2 \text{ is active} \\ 0 & \text{if } I \text{ is active and } F_2 \text{ is active} \\ 1 & \text{if } I \text{ is inactive and } F_2 \text{ is inactive.} \end{cases} \tag{A17}$$

The equilibrium activities of x_i are as follows. If I is active and F_2 is inactive, then

$$x_i \cong \frac{I_i}{1 + A_1 I_i + A_1 B_1 + C_1}. \tag{A18}$$

Thus $x_i > 0$ iff $I_i > 0$. If I is inactive and F_2 is active, then (A5) holds. If I is active and F_2 is active, then (A5) holds. If I is inactive and F_2 is inactive, then

$$x_i \cong \frac{B_1 - B_1}{1 + A_1 B_1 + C_1} = 0. \tag{A19}$$

Hence no output signals are generated from F_1. The coefficient B_1 in term $B_1 G_3$ of (A16) may be decreased somewhat without changing system dynamics.

In Fig. 14d, the gain control cells are phasically excited by bottom–up signals and inhibited by top–down signals. Once active, they nonspecifically excite F_1 cells. In the absence of any external signals, F_1 cells are maintained in a state of tonic hyperpolarization, or negativity. In response to a bottom–up input, the gain control channel balances the tonic hyperpolarization of F_1 cells, thereby allowing those cells which receive bottom–up inputs to fire. When a top–down signal is active, no gain control outputs occur. Hence top–down signals alone cannot overcome the tonic hyperpolarization enough to generate output signals from F_1. Simultaneous convergence of an excitatory bottom–up signal and an inhibitory top–down signal at the gain control cells prevents these cells from generating output signals to F_1. Consequently, only those F_1 cells at which a bottom–up input and top–down template signal converge can overcome the tonic hyperpolarization to generate output signals.

The STM equations of F_1 are

$$\varepsilon \frac{dx_i}{dt} = -x_i + (1 - A_1 x_i)\left(I_i + D_1 \sum_j f(x_j) z_{ji} + B_1 G_4\right) - (B_1 + C_1 x_i), \tag{A20}$$

where

$$G_4 = \begin{cases} 1 & \text{if } I \text{ is active and } F_2 \text{ is inactive} \\ 0 & \text{if } I \text{ is inactive and } F_2 \text{ is active} \\ 0 & \text{if } I \text{ is active and } F_2 \text{ is active} \\ 0 & \text{if } I \text{ is inactive and } F_2 \text{ is inactive.} \end{cases} \tag{A21}$$

The equilibrium activities of x_i are as follows. If I is active and F_2 is inactive, then

(A18) holds. If I is inactive and F_2 is active, then (A5) holds. Equation (A5) also holds if I is active and F_2 is active. If I is inactive and F_2 is inactive, then (A15) holds.

In all four cases, an F_1 cell fires only if the number of active excitatory pathways which converge upon the cell exceeds the number of active inhibitory pathways which converge upon the cell, where we count a source of tonic hyperpolarization as one input pathway. A similar rule governs the firing of the gain control channel in all cases.

ACKNOWLEDGMENTS

Thanks to Cynthia Suchta and Carol Yanakakis for their valuable assistance in the preparation of the manuscript.

REFERENCES

1. E. Basar, H. Flohr, H. Haken, and A. J. Mandell (Eds.), *Synergetics of the Brain*, Springer-Verlag, New York, 1983.
2. J. P. Banquet and S. Grossberg, Probing cognitive processes through the structure of event-related potentials during learning: An experimental and theoretical analysis, Submitted for publication, 1986.
3. G. A. Carpenter and S. Grossberg, Category learning and adaptive pattern recognition: A neural network model, Proceedings of the Third Army Conference on Applied Mathematics and Computing, ARO Report 86-1, 1985, 37–56.
4. G. A. Carpenter and S. Grossberg, Neural dynamics of adaptive pattern recognition: Priming, search, attention, and category formation, *Soc. Neurosci. Abstracts*, **11**, 1985, 1110.
5. G. A. Carpenter and S. Grossberg, Neural dynamics of category learning and recognition: Attention, memory consolidation, and amnesia, in *Brain Structure, Learning, and Memory* (J. Davis, R. Newburgh, and E. Wegman, Eds.), AAAS Symposium Series, 1986.
6. G. A. Carpenter and S. Grossberg, Neural dynamics of category learning and recognition: Structural invariants, reinforcement, and evoked potentials, in *Pattern Recognition and Concepts in Animals, People, and Machines* (M. L. Commons, S. M. Kosslyn, and R. J. Herrnstein, Eds.), Erlbaum, Hillsdale, NJ, 1986.
7. S. Grossberg, Adaptive pattern classification and universal recoding. I. Parallel development and coding of neural feature detectors, *Biol. Cybernet.* **23**, 1976, 121–134.
8. S. Grossberg, Adaptive pattern classification and universal recoding. II. Feedback, expectation, olfacation, and illusions, *Biol. Cybernet.* **23**, 1976, 187–202.
9. S. Grossberg, A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans, in *Progress in Theoretical Biology* (R. Rosen and F. Snell, Eds.), Vol. 5, pp. 233–374, Academic Press, New York, 1978.
10. S. Grossberg, How does a brain build a cognitive code? *Psychol. Rev.*, **87**, 1980, 1–51.
11. S. Grossberg, The adaptive self-organization of serial order in behavior: Speech, language, and motor control, in *Pattern Recognition by Humans and Machines* (E. C. Schwab and H. C. Nusbaum, Eds.), Vol. 1, Academic Press, New York, 1986.
12. S. Grossberg and G. O. Stone, Neural dynamics of word recognition and recall: Attentional priming, learning, and resonance, *Psychol. Rev.* **93**, 1986, 46–74.
13. S. Grossberg and G. O. Stone, Neural dynamic of attention switching and temporal order information in short term memory, *Memory and Cognition*, in press, 1986.
14. S. Grossberg, Do all neural networks really look alike? A comment on Anderson, Silverstein, Ritz, and Jones, *Psychol. Rev.* **85**, 1978, 592–596.
15. J. A. Anderson, J. W. Silverstein, S. R. Ritz, and R. S. Jones, Distinctive features, categorical perception, and probability learning: some applications of a neural model, *Psychol. Rev.* **84**, 1977, 413–451.
16. K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, *Biol. Cybernet.* **36**, 1980, 193–202.
17. J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Nat. Acad. Sci. U.S.A.* **79**, 1982, 2554–2558.

18. T. Kohonen, *Associative Memory: A System-Theoretical Approach*, Springer-Verlag, New York, 1977.
19. J. I. McClelland and D. E. Rumelhart, Distributed memory and the representation of general and specific information, *J. Exp. Psychol. Gen.* **114**, 1985, 159–188.
20. M. I. Posner, *Cognition: An Introduction*, Scott, Foresman, Glenview, Ill., 1973.
21. E. E. Smith and D. L. Medin, *Categories and Concepts*, Harvard Univ. Press, Cambridge, Mass., 1981.
22. M. I. Posner and S. W. Keele, On the genesis of abstract ideas, *J. Exp. Psychol.* **77**, 1968, 353–363.
23. M. I. Posner and S. W. Keele, Retention of abstract ideas, *J. Exp. Psychol.* **83**, 1970, 304–308.
24. M. A. Cohen and S. Grossberg, Neural dynamics of speech and language coding: Developmental programs, perceptual grouping, and competition for short term memory, *Human Neurobiology*, 1986, **5**, 1–22.
25. M. A. Cohen and S. Grossberg, Unitized recognition codes for parts and wholes: The unique cue in configural discriminations, in *Pattern Recognition and Concepts in Animals, People, and Machines* (M. L. Commons, S. M. Kosslyn, and R. J. Herrnstein, Eds.), Erlbaum, Hillsdale, N.J., 1986.
26. M. A. Cohen and S. Grossberg, Masking fields: A massively parallel architecture for discovering, learning, and recognizing multiple groupings of patterned data. *Applied Optics*, in press, 1986.
27. S. Grossberg, Some psychophysiological and pharmacological correlates of a developmental, cognitive and motivational theory, in *Brain and Information: Event Related Potentials* (R. Karrer, J. Cohen, and P. Tueting, Eds.), New York Academy of Sciences, New York, 1984, pp. 58–151.
28. S. Grossberg, The quantized geometry of visual space: The coherent computation of depth, form, and lightness, *Behavioral Brain Sci.* **6**, 1983, 625–692.
29. C. C. Lin and L. A. Segal, *Mathematics Applied to Deterministic Problems in the Natural Sciences*, Macmillan, New York, 1974.
30. S. Ellias and S. Grossberg, Pattern formation, contrast control, and oscillations in the short term memory of shunting on-center off-surround networks, *Biol. Cybernet.* **20**, 1975, 69–98.
31. S. Grossberg, Contour enhancement, short-term memory, and constancies in reverberating neural networks, *Studies Appl. Math.*, **52**, 1973, 217–257.
32. S. Grossberg and D. Levine, Some developmental and attentional biases in the contrast enhancement and short term memory of recurrent neural networks, *J. Theoret. Biol.* **53**, 1975, 341–380.
33. G. A. Carpenter, and S. Grossberg, Self-organization of neural recognition codes: Nonlinear Weber Law modulation of associative learning, 1986, in preparation.
34. C. von der Malsburg and D. J. Willshaw, Differential equations for the development of topological nerve fibre projections, in *Mathematical Psychology and Psychophysiology* (S. Grossberg, Ed.), Amer. Math. Soc., Providence, R.I., 1981, pp. 39–47.
35. D. O. Hebb, *The Organization of Behavior*, Wiley, New York, 1949.
36. S. Grossberg, Unitization, automaticity, temporal order, and word recognition. *Cognition Brain Theory*, **7**, 1984, 263–283.
37. S. Grossberg, Cortical dynamics of three-dimensional form, color, and brightness perception: Parts I and II. *Perception Psychophys.*, in press, 1986.
38. S. Grossberg, and E. Mingolla, Neural dynamics of form perception: Boundary completion, illusory figures, and neon color spreading, *Psychol. Rev.*, **92**, 1985, 173–211.
39. S. Grossberg and E. Mingolla, Neural dynamics of perceptual grouping: Textures, boundaries, and emergent segmentations. *Perception Psychophys.*, **38**, 1985, 141–171.
40. S. Grossberg and E. Mingolla, Neural dynamics of surface perception: Boundary webs, illuminants, and shape-from-shading, *Comput. Vision, Graphics, Image Process.*, **37**, (1987) 116–165.