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. © 1987 Academic Press, Inc.

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

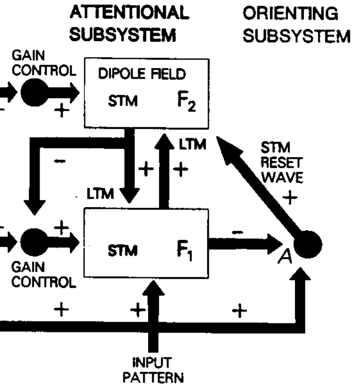

FIG. 1. Anatomy of the attentional-orienting system: Two successive stages, F_1 and F_2, of the attentional subsystem encode patterns of activation in short term memory (STM). Bottom–up and top–down pathways between F_1 and F_2 contain adaptive long term memory (LTM) traces which multiply the signals 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*

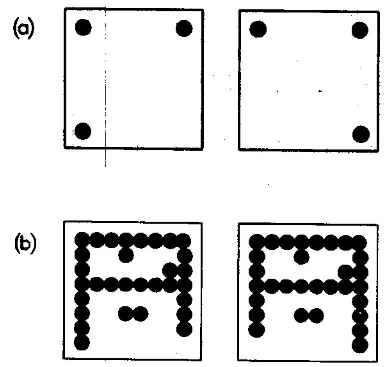

FIG. 2. Self-scaling property discovers critical features in a context-sensitive way: (a) Two input patterns of 3 features mismatch at 1 feature. When this mismatch is sufficient to generate distinct recognition codes for the two patterns, the mismatched features are encoded in LTM as part of the critical 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

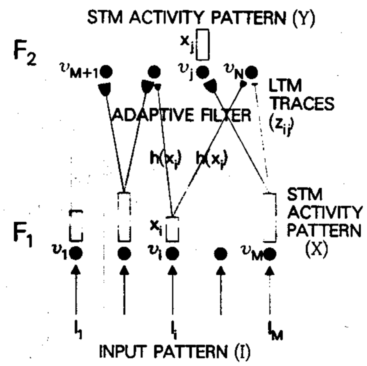

FIG. 3. Stages of bottom–up activation: The input pattern I generates a pattern of STM activation X across F_1. Sufficiently active F_1 nodes emit bottom–up signals to F_2. This signal pattern S is gated by long term memory (LTM) traces within the $F_1 \to F_2$ pathways. The LTM-gated signals are summed 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}.

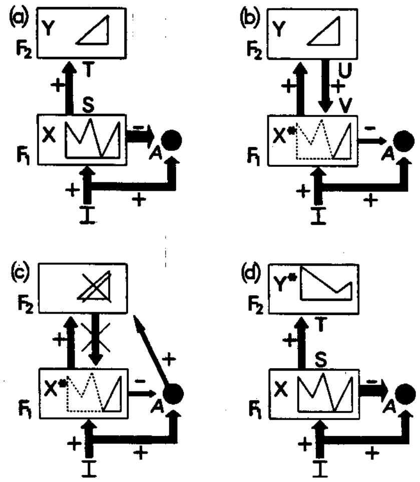

FIG. 4. Search for a correct F_2 code: (a) The input pattern I generates the specific STM activity pattern X at F_1 as it nonspecifically activates A. Pattern X both inhibits A and generates the output signal pattern S. Signal pattern S is transformed into the input pattern T, which activates the STM 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, corresponding to a particular node (or "pointer," or "index") in F_2. Su categorical mechanism is both interesting in itself and a necessary prelude to analysis of recognition codes in which multiple groupings of X are simultane represented by Y. In the example that is characterized in this article, level designed to make a choice.

All the LTM traces in the adaptive filter, and thus all learned past experienc the network, are used to determine the recognition code Y via the transform $I \to X \to S \to T \to Y$. However, only those nodes of F_2 which maintain st activity in the STM pattern Y can elicit new learning at contiguous LTM tr Because the recognition code Y is a more contrast-enhanced pattern than T, F_2 nodes which receive positive inputs ($I \to X \to S \to T$) may not store any activity ($T \to Y$). The LTM traces in pathways leading to these nodes thus influ the recognition event but are not altered by the recognition event. Some mem 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 $X \to Y$ takes place, the activities Y in F_2 elicit a top–down excitatory signal pattern U back to F_1 (Fig. Only sufficiently large STM activities in Y elicit signals in U along the feed pathways $F_2 \to F_1$. As in the bottom–up adaptive filter, the top–down signals also gated by LTM traces and the LTM-gated signals are summed at F_1 nodes. pattern U of output signals from F_2 hereby generates a pattern V of LTM-gated summed input signals to F_1. The transformation from U to V is thus als adaptive filter. The pattern V is called a *top–down template*, or *learned expecta*

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

The entire activation sequence

$$I \to X \to S \to T \to Y \to U \to V \to X^*$$

takes place very quickly relative to the rate with which the LTM traces in eithe bottom–up adaptive filter $S \to T$ or the top–down adaptive filter $U \to V$ change. Even though none of the LTM traces changes during such a short their prior learning strongly influences the STM patterns Y and X^* that e within the network by determining the transformations $S \to T$ and $U \to V$. We discuss how a match or mismatch of I and V at F_1 regulates the course of lea in response to the pattern I, and in particular solves the stability-plasticity dile (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. Th input pattern I also excites the orienting subsystem A, but pattern X at F_1 inhibit A before it can generate an output signal. Activity pattern X also elicits an outpu pattern S which, via the bottom–up adaptive filter, instates an STM activity patter 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 acros F_1. The amount by which activity in X is attenuated to generate X^* depends upo 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 th inhibitory signal from F_1 to A is also attenuated. If the attenuation is sufficientl great, inhibition from F_1 to A can no longer prevent the arousal source A fron firing. Fig. 4c depicts how disinhibition of A releases an arousal burst to F_2 whicl equally, or nonspecifically, excites all the F_2 cells. The cell populations of F_2 react t such an arousal signal in a state-dependent fashion. In the special case that F 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 physiologi cal 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 gate by habituating chemical transmitters. A nonspecific arousal burst induces selectiv 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, an thereby terminates the mismatch between I and V. Input pattern I can thu reinstate the original activity pattern X across F_1, which again generates the outpu pattern S from F_1 and the input pattern T to F_2. Due to the enduring inhibition a F_2, the input pattern T can no longer activate the original pattern Y at F_2. A nev pattern Y^* is thus generated at F_2 by I (Fig. 4d). Despite the fact that some F nodes may remain inhibited by the STM reset property, the new pattern Y^* ma encode large STM activities. This is because level F_2 is designed so that its tota suprathreshold activity remains approximately constant, or normalized, despite th fact that some of its nodes may remain inhibited by the STM reset mechanism. Thi property is related to the limited capacity of STM. A physiological process capabl of achieving the STM normalization property is based upon on-center off-surroun feedback interactions among cells obeying membrane equations [10, 28].

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

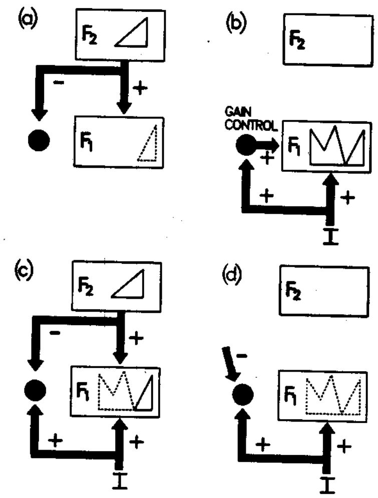

FIG. 5. Matching by the $\frac{2}{3}$ Rule: (a) A top–down template from F_2 inhibits the attentiona control source as it subliminally primes target F_1 cells. (b) Only F_1 cells that receive bottom–up i and gain control signals can become supraliminally active. (c) When a bottom–up input pattern top–down template are simultaneously active, only those F_1 cells that receive inputs from both sc can become supraliminally active. (d) Intermodality inhibition can shut off the F_1 gain control s and thereby prevent a bottom–up input from supraliminally activating F_1. Similarly, disinhibition 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 acros reads out a top–down template which matches I, to the degree of accuracy requ by the level of attentional vigilance (Sect. 2D), or which has not yet undergone prior learning. In the latter case, a new recognition category is then established 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 derive considering its role in the regulation of attentional priming. Consider, for examp situation in which F_2 is activated by a level other than F_1 before F_1 car activated by a bottom–up input (Fig. 5a). In such a situation, F_2 can genera top–down template V to F_1. The level F_1 is then primed, or sensitized, to recei bottom–up input that may or may not match the active expectancy. As depicte Fig. 5a, level F_1 can be primed to receive a bottom–up input without necess eliciting suprathreshold output signals in response to the priming expectancy.

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

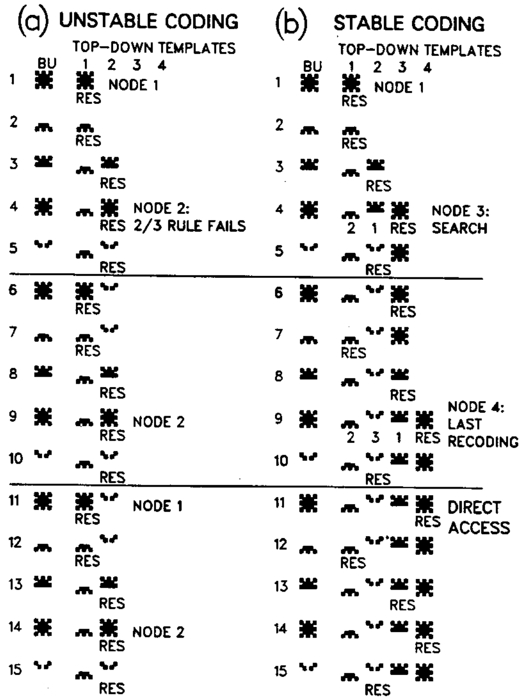

FIG. 6. Stabilization of categorical learning by the $\frac{2}{3}$ Rule: In both (a) and (b), four input patterns A, B, C, and D are presented repeatedly in the list order $ABCAD$. In (a), the $\frac{2}{3}$ Rule is violated because the top–down inhibitory gain control mechanism is weak (Fig. 5c). Pattern 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

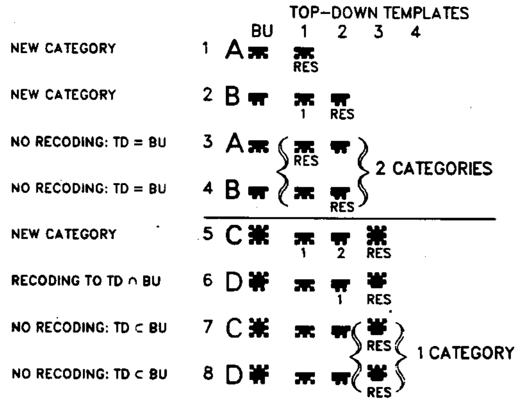

FIG. 7. Distinguishing noise from patterns for inputs of variable complexity: Input patterns A and B are coded by the distinct category nodes v_{M+1} and v_{M+2}, respectively. Input patterns C and D include A and B as subsets, but also possess identical subpatterns of additional features. Due to this 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

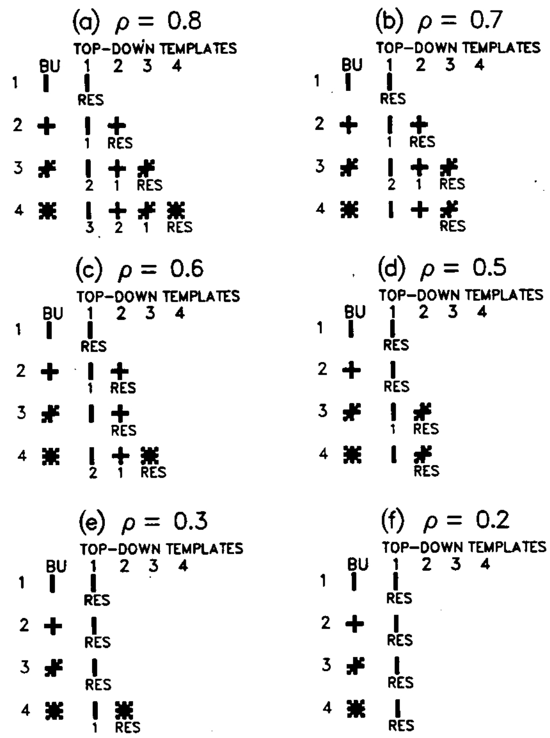

FIG. 8. Influence of vigilance level on categorical groupings: As the vigilance parameter ρ decreases, the number of categories progressively decreases.

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.

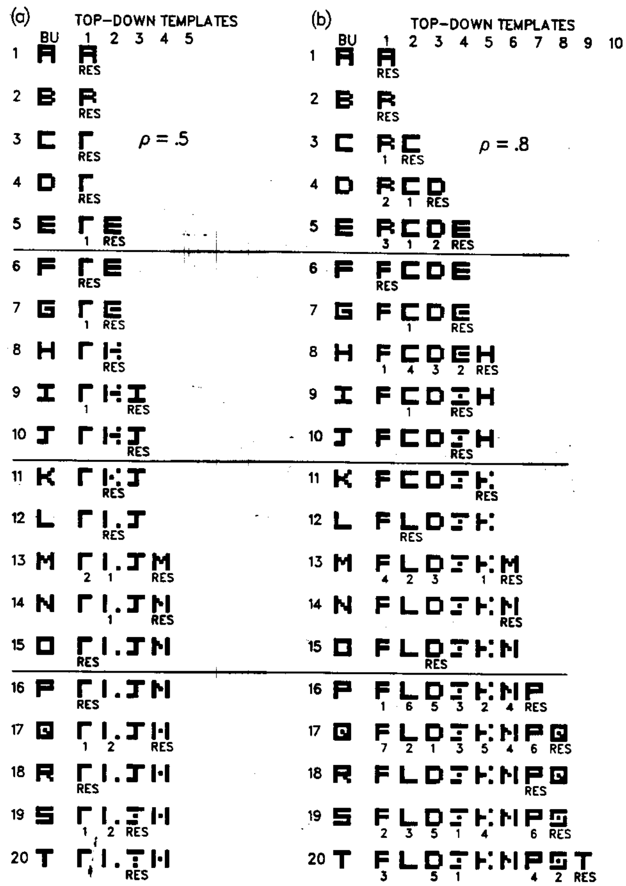

FIG. 9. Alphabet learning: different vigilance levels cause different numbers of letter categories and different critical feature patterns, or templates, to form.

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

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

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.

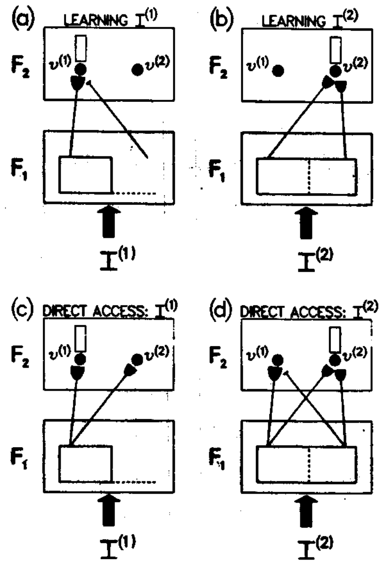

FIG. 10. The Weber Law Rule and the Associative Decay Rule enable both subset and superset input patterns to directly access distinct F_2 nodes: (a) and (b) schematize the learning induced by presentation of $I^{(1)}$ (a subset pattern) and $I^{(2)}$ (a superset pattern). Larger path endings designate 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 suprathreshold. 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

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

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,

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

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 $\mathbf{I}$ 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 $\mathbf{I}$ 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 $\mathbf{X}$ 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 $\sum_{i \in \mathbf{I}} z_{ij}$ 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 subsec theorems in the article assume that these hypotheses hold.

Binary Input Patterns

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

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

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

Weber Law Rule and Bottom–Up Associative Decay Rule

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

Template Learning Rule and Top–Down Associative Decay Rule

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

Reset Rule

An active F_2 node v_j is reset if

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

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 pr learning trial, then

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

where

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

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 sa the hypotheses in Section 18. We will begin by characterizing the order of se This analysis provides a basis for proving that learning self-stabilizes and lea recognition by direct access.

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

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

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

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

When an input I is first presented on a trial,

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

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

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

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

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

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

Order Function

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

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),

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

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 ch
of the bottom–up filter.

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

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

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

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

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

even though the full ordering of the T_j's may be different from that defined by
O_j's. If v_{j_1} is reset by the orienting subsystem, then template $V^{(j_1)}$ is shut off fo
remainder of the trial and subsequent values of T_{j_1} do not influence which F_2 n
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 $O_{j_2} > O_{j_3} >$
node v_{j_2} is chosen next and template $V^{(j_2)}$ is read-out. Theorem 2 implies that

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

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

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

20. STABLE CATEGORY LEARNING

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

THEOREM 4 (Learning on a single trial). *Assume the model hypotheses of Sec tion 18. Suppose that an F_2 node v_J is chosen for* STM *storage and that read-out of th 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 increase and all other T_j remain constant, thereby confirming the choice of v_J by the adaptiv filter. In addition, the set $\mathbf{I} \cap \mathbf{V}^{(J)}$ remains constant during learning, so that learnin 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 monotonicall 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}. Usin these facts, we then show that the $z_{iJ}(t)$ change monotonically, that $T_J(t)$ can onl 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 doe 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 activitie 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, whil v_J is active. We suppose first that v_J is not an uncommitted node before considerin the case in which v_J is an uncommitted node. While v_J remains active, the se $\mathbf{X}(t) = \mathbf{I} \cap \mathbf{V}^{(J)}(t_0)$. 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 fac 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