Boston University

OpenBU http://open.bu.edu

Cognitive & Neural Systems CAS/CNS Technical Reports

1994-03

Store Working Memory Networks for Storage and Recall of Arbitrary Temporal Sequences

https://hdl.handle.net/2144/2108

Boston University

STORE WORKING MEMORY NETWORKS FOR STORAGE AND RECALL OF ARBITRARY TEMPORAL SEQUENCES

Gary Bradski, Gail A. Carpenter, and Stephen Grossberg

September, 1992

Revised: March 1994

Technical Report CAS/CNS-92-028

Boston University Center for Adaptive Systems and
Department of Cognitive and Neural Systems
111 Cummington Street
Boston, MA 02215

STORE WORKING MEMORY NETWORKS FOR STORAGE AND RECALL OF ARBITRARY TEMPORAL SEQUENCES

Gary Bradski†, Gail A. Carpenter‡, and Stephen Grossberg§
Center for Adaptive Systems
and
Department of Cognitive and Neural Systems
Boston University
111 Cummington Street
Boston, Massachusetts 02215

September, 1992

Revised: March 25, 1994

Technical Report CAS/CNS-TR-92-028
Boston, MA: Boston University

Send requests for reprints to:
Professor Stephen Grossberg
Boston University
Center for Adaptive Systems
111 Cummington Street
Boston, MA 02215

† Supported in part by the Air Force Office of Scientific Research (AFOSR 90-0128) and the Office of Naval Research (ONR N00014-91-J-4100 and ONR N00014-92-J-1309).

‡ Supported in part by British Petroleum (BP 89A-1204), ARPA (AFOSR 90-0083 and ONR N00014-92-J-4015), the National Science Foundation (NSF IRI-90-00530), and the Office of Naval Research (ONR N00014-91-J-4100).

§ Supported in part by the Air Force Office of Scientific Research (AFOSR F49620-92-J-0225), ARPA (AFOSR 90-0083 and ONR N00014-92-J-4015) and the Office of Naval Research (ONR N00014-91-J-4100 and ONR N00014-92-J-1309).

Acknowledgements: The authors wish to thank Cynthia E. Bradford and Diana Meyers for their valuable assistance in the preparation of the manuscript.

ABSTRACT

Neural network models of working memory, called **S**ustained **T**emporal **O**rder **RE**current (STORE) models, are described. They encode the invariant temporal order of sequential events in short term memory (STM) in a way that mimics cognitive data about working memory, including primacy, recency, and bowed order and error gradients. As new items are presented, the pattern of previously stored items is invariant in the sense that relative activations remain constant through time. This invariant temporal order code enables all possible groupings of sequential events to be stably learned and remembered in real time, even as new events perturb the system. Such a competence is needed to design self-organizing temporal recognition and planning systems in which any subsequence of events may need to be categorized in order to to control and predict future behavior or external events. STORE models show how arbitrary event sequences may be invariantly stored, including repeated events. A preprocessor interacts with the working memory to represent event repeats in spatially separate locations. It is shown why at least two processing levels are needed to invariantly store events presented with variable durations and interstimulus intervals. It is also shown how network parameters control the type and shape of primacy, recency, or bowed temporal order gradients that will be stored.

1. Introduction: STORE Working Memory Models

Working memory is a kind of short term memory (STM) whereby a temporally ordered sequence of events can be temporarily stored (Baddeley, 1986). Events that are stored in working memory may be sequentially recalled, or quickly erased by a distracting event, in contrast to long term memory (LTM). A large experimental literature and a variety of models elucidate the properties of working memory (Atkinson and Shiffrin, 1971; Elman, 1990; Grossberg, 1970; Grossberg and Pepe, 1971; Gutfreund and Mezard, 1988; Guyon, Personnaz, Nadal, and Dreyfus, 1988; Jordan, 1986; Reeves and Sperling, 1986; Schreter and Pfeifer, 1989; Seibert, 1991; Seibert and Waxman, 1990a, 1990b; Wang and Arbib, 1990).

A class of dynamically defined working memory neural network models, called **S**ustained **T**emporal **O**rder **RE**current (STORE) models, encode the temporal order of arbitrary sequences of items. Larger STM activations are recalled first, and hence represent earlier items. The ratio of STM codes of previous inputs remains constant as new inputs enter working memory, even when input durations and interstimulus intervals vary widely. This invariance property allows all possible groupings of sequential events to be stably learned and remembered in real time, because invariant activity ratios imply a learnable invariance of recognition codes in competitive learning or self-organizing feature map models that receive their inputs from a STORE model. STORE models thus realize an Invariance Principle (Grossberg, 1978a, 1978b) that enables chunks (compressed, categorical, or unitized representations) of variable size to be encoded in LTM in a manner that is not destabilized as new items are added to previously learned sequences. Grossberg (1978a, 1978b) proved that the Invariance Principle implies that items are not always stored in working memory with veridical temporal order. Thus the fundamental constraint that temporal learning be stable implies that model working memories, like those of humans, do not always encode information in correct temporal order. Correspondingly, a large cognitive database can be explained by STORE models, as noted in Sections 2–4.

Figure 1

The basic, two-level model (STORE 1) that is described in Section 2 encodes temporal order for input sequences whose items are not repeated (Bradski, Carpenter, and Grossberg, 1991, 1992). This paper develops two extensions of the STORE 1 model. First, an STM decay term in the STORE 2 class of models adds a parametric degree of freedom to the control of relative sizes of working memory representations (Section 6). This physically important parameter facilitates the quantitative modeling of cognitive data. Another generalization of the model (STORE 3) extends system capabilities by allowing both repeated and non-repeated item sequences to be encoded and recalled (Section 7). This is accomplished using either a Winner-Take-All (WTA) or a Positional Gradient Shift (PGS) preprocessor. Each preprocessor causes spatially distinct network nodes to become active when an input item is repeated. This separation allows the network to invariantly store arbitrary sequences in working memory. In addition, a simplified, one-level model (STORE 0) is described and shown to be adequate for working memory coding and recall, provided that input durations are restricted (Section 5). This one-level model clarifies why two levels are needed to invariantly store items of variable duration. Section 8 includes other variants of the STORE model that illustrate the flexibility and scope of the STORE design. Section 9 describes applications of STORE models to temporal recognition, planning, and inference problems.

2. Invariance Principle and Normalization Rule

The STORE neural network working memories are based upon algebraically characterized working memories that were introduced by Grossberg (1978a, 1978b). These algebraic working memories were designed to explain psychological data concerning working memory storage and recall. In these models, individual events are stored in working memory in such a way that the pattern of STM activity across event representations encodes both the events that have occurred and the temporal order in which they have occurred. In psychological terms, the working memory stores both *item* information and *order* information (Healy, 1975; Lee and Estes, 1981; Ratcliff, 1978). The models also include a mechanism for reading out events in the stored temporal order. Relative activation strengths translate into order of performance. A nonspecific rehearsal wave opens a gate to read out stored activities. After rehearsal begins, the most active node reaches its output threshold first, then self-inhibits its activation via a negative feedback pathway to enable the next most active node to be rehearsed, and so on, until all active nodes are reset. An event sequence can hereby be performed from STM even if it is not yet incorporated through learning into LTM, much as a new telephone number can be repeated the first time that it is heard.

The large data base on working memory shows that storage and performance of temporal order information from working memory is not always veridical (Atkinson and Shiffrin, 1971; Baddeley, 1986; Reeves and Sperling, 1986). These deviations from veridical temporal order in STM were given an explanation by the algebraic working memory model as consequences of two design principles that have clear adaptive value. These principles are called the Invariance Principle and the Normalization Rule (Grossberg, 1978a, 1978b).

2.1. Invariance Principle: The spatial patterns of STM activation across the event representations of a working memory are stored and reset in response to sequentially presented events so as to leave the temporal order codes of all past event groupings invariant. In particular, a temporal list of events in STM preserves the stability of previously learned LTM codes for familiar sublists of the list. For example, suppose that the word MY has previously been stored in a working memory's STM and has established a learned chunk in LTM. Suppose that the word MYSELF is then stored for the first time in STM. The STM encoding of MY as a syllable of MYSELF may not be the same as its STM encoding as a word. On the other hand, MY's STM encoding as part of MYSELF should not cause forgetting of the LTM code for MY as a word. If it did, familiar words, such as MY, could not be learned as parts of larger words, such as MYSELF, without eliminating the smaller words from the lexicon. More generally, new wholes could not be built from familiar parts without erasing LTM of the parts.

The Invariance Principle can be algebraically realized as follows, provided that no list items are repeated. Assume for simplicity that the i^{th} list item is preprocessed by a winner-take-all network. Each list item then activates a single output node of the preprocessor network. Properties of the working memory also hold if a finite set of output nodes is activated for each item. The winner-take-all case is described herein for notational simplicity. Let the winner-take-all node that is activated by the i^{th} item send a binary input I_i to the first working memory level F_1. Let x_i denote the activity of the i^{th} item representation of F_1. Suppose that I_i is registered in working memory at time t_i. At time t_i, the activity pattern $(x_1(t_i), x_2(t_i), \ldots, x_n(t_i))$ across F_1 stores the effects of the list $I_1, I_2, \ldots, I_i$ of previous inputs. The input I_i updates the activity values $x_k(t_{i-1})$ to new values $x_k(t_i)$ for all nodes $k = 1, 2, \ldots, i$ according to the following rule: At time t_i, the pattern $(x_1(t_{i-1}), x_2(t_{i-1}), \ldots, x_{i-1}(t_{i-1}))$ of previously stored STM activities is multiplied by a common factor ω_i as the i^{th} item is instated with some activity μ_i.

This storage rule satisfies the Invariance Principle for the following reason. Suppose that F_1 is the first level of a two-level competitive learning network (Grossberg, 1976). Then F_1 sends signals to the second level F_2 via an adaptive filter. The total input to the j^{th} F_2 node is $\Sigma_k x_k z_{kj}$, where z_{kj} denotes the LTM trace, or adaptive weight, in the path from the k^{th} F_1 node to the j^{th} F_2 node. In psychological terms, each active F_2 node represents a chunk of the F_1 activity pattern. When the j^{th} F_2 node is active, the LTM weights z_{kj} converge toward x_k; in other words, the weight vector becomes parallel to the F_1 activity vector. When a new item is added to the list, the Invariance Principle implies that the previously active items in the list will simply be multiplied by a common factor, thereby maintaining a constant ratio between the previously active items. Constant activity ratios imply that the former F_1 activity vector remains parallel to its weight vector as its magnitude changes under new inputs. Hence, adding new list items does not invalidate the STM and LTM codes for sublists. In particular, the temporal order of items in each sublist, encoded as relative sizes of both the STM and the LTM variables, remains invariant.

2.2. Normalization Rule: The Normalization Rule algebraically instates the classical property of the limited capacity of STM (Atkinson and Shiffrin, 1971). According to this property, the total network STM activity across all nodes can equal, or increase to, a finite maximum value S that is insensitive to the total number of active nodes; hence, is normalized. Parameter S characterizes the "limited capacity" of STM. In human subjects, this parameter is determined by biological constraints. In an artificial neural network, parameter S can be set at any finite value.

3. Relation to Speech and Language Data

The algebraic Invariance Principle and Normalization Rule imply (Grossberg, 1978b) that the pattern $(x_1, \ldots, x_i)$ of stored STM activities can exhibit primacy (all $x_{k-1} > x_k$), recency (all $x_{k-1} < x_k$), or bowing, which combines primacy for early items with recency for later items (Figure 1c). Primacy, recency, and bowing correspond to properties of STM storage by human subjects. Model parameters are typically set so that the STM activity pattern exhibits a primacy gradient in response to a short list. Since more active nodes are read out of STM before less active nodes during performance trials, primacy storage leads to the correct order of recall in response to a short list. Using the same parameters, the STM activity pattern exhibits a bow in response to longer lists, and approaches a recency gradient in response to still longer lists. An STM bow leads to performance of items near the list beginning and end before items near the list middle. A larger STM activity at a node also leads to a higher probability of recall from that node under circumstances when the network is perturbed by noise. An STM bow thus leads to earlier recall and to a higher probability of recall from items at the beginning and the end of a list.

These formal network properties are also properties of data from a variety of experiments about working memory, such as free recall experiments during which human subjects are asked to recall list items after being exposed to them once in a prescribed order (Atkinson and Shiffrin, 1971; Healy, 1975; Lee and Estes, 1981). Effects of LTM on free recall data have also been analysed by the theory (Grossberg, 1978a, 1978b), as have reaction time data from experiments about the sequential performance of stored motor commands (Boardman and Bullock, 1991), data concerning errors in serial item and order recall due to rapid attention shifts (Grossberg and Stone, 1986a), data concerning errors and reaction times during lexical priming and episodic memory experiments (Grossberg and Stone, 1986b), and data concerning word superiority, phonemic restoration, and backward effects on speech perception (Cohen and Grossberg, 1986; Grossberg, 1986). These data explanations provide converging evidence that working memory models which satisfy STORE design principles are used in the brain. The present article extends the computational capabilities of this class of models.

4. The Basic Model: STORE 1

In Bradski, Carpenter, and Grossberg (1992), we showed how neural networks could be defined which store invariant and normalized activation patterns in working memory. These activation patterns are emergent properties of the network dynamics, rather than formal algebraic rules. Such a step is needed to encode complex events that may be occurring asynchronously in time, as well as to design hierarchies of working memories $W_1, W_2, \ldots, W_n, \ldots$, such that each node of W_n codes a compressed representation of a stored activation pattern across the working memory W_{n-1}. The nodes of each successive W_n code "higher invariants" or "chunks" of the items coded by W_1.

The working memory model STORE 1 that was defined in Bradski, Carpenter, and Grossberg (1992) is a two-layer input-gated neural network (Figure 1a). The first layer (F_1) is a competitive system, whose activity vector $(x_1, x_2, \ldots, x_n)$ represents working memory. The second layer (F_2) tracks and stores the STM activity of the first layer via its activity vector $(y_1, y_2, \ldots, y_n)$. Inputs are presented as a sequence of non-repeated items, with arbitrary intra-input durations α_i and inter-input durations β_i (Figure 1b). The ith input to the STORE 1 system consists of a unit input I_i from the ith node of the input field F_0. Input I_i may represent activation of a recognition category that results from compressing a distributed representation of an individual event, or item, at an earlier processing level. The STORE input vector $\mathbf{I}$ then represents STM activity of a winner-take-all field (F_0) that categorizes previously learned item recognition codes with a normalized activity. That is why inputs I_i are chosen equal to 0 or 1. The STORE working memory responds to these normalized inputs by storing the temporal order of item representations.

After entering working memory, items stored at F_1 are recalled in the order of their STM activities x_k, from largest to smallest. When system parameters are set so that F_1 stores a primacy gradient (Figure 1c), therefore, items are recalled in the order in which they were presented. Other parameter ranges yield patterns of bowing or recency in STM. The dimensionless equations (1)–(3) describe the input and STM of a STORE 1 system (Figure 1).

STORE 1: F_0 Input

$$I_i(t) = \begin{cases} 1 & \text{if } \alpha_i - t_i < t < t_i \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

STORE 1: F_1 Working Memory

$$\frac{dx_i}{dt} = (AI_i + y_i - x_i x)I, \tag{2}$$

where $x \equiv \sum_k x_k$ and $I \equiv \sum_k I_k$.

STORE 1: F_2 Stored Memory

$$\frac{dy_i}{dt} = (x_i - y_i)I^c, \tag{3}$$

where $I^c \equiv 1 - I$. Initially, $x_i(0) = y_i(0) = 0$.

Analysis of STORE 1 (Bradski, Carpenter, and Grossberg, 1992) shows that the STM pattern at F_1 stores a (veridical) primacy gradient if parameter A is small; that bowing can occur if $0 < A < 1$; and that F_1 stores a recency gradient if $A \geq 1$. These conclusions hold under the assumption that the F_1 STM variables x_k relax to their steady-state values during each input presentation interval $(t_i - \alpha_i, t_i)$, when $I = 1$ in equation (2); and that the F_2 STM variables y_k relax to their steady-state values during each inter-input interval $[t_i, t_i + \beta_i]$, when $I^c = 1$ in equation (3) (Figure 1b). In a typical STORE 1 simulation, input durations were randomly varied between 10 and 40, with the input intervals $(t_i - t_{i-1})$ set equal to 50. Input duration variations do not affect the stored activity pattern. Insight into how STORE 1 works is provided in terms of a mathematical analysis of the more general STORE 2 model (Section 6). In particular, the nonspecific gain, or gating, term I in (2) enables the working memory activities to respond to inputs I_i while they are on, since $I = 1$ if any $I_i = 1$. The complementary gating term I^c in (3) prevents the stored memories y_i from responding to inputs I_i while they are on, since $I^c = 0$ if any $I_i = 1$. Already stored activities y_i are hereby buffered against distortion by future inputs $I_j, j > i$. Each stored activity y_i also influences its working memory activity x_i via (2), and thus the inhibitory effect of total activity x on how strongly x_j is activated by $I_j, j > i$.

The constraint that x_i and y_i can approach their new equilibria in response to I_i requires that the input presentation interval α_i and the inter-input interval β_i (Figure 1b) both be positive; infinitely fast presentation rates, with $\alpha_i = \beta_i \cong 0$ are not admissible. The input intervals α_i and β_i may be arbitrarily small, however, provided that the rates with which x_i and y_i react are chosen large enough. Given fixed rates, the model exhibits a fastest input presentation rate beyond which successive events cannot be resolved, as is also seen in brain data (Miller, 1981; Miller and Liberman, 1979; Repp, Liberman, Eccardt, and Pestsky, 1978; Tarttar, Kat, Samuel, and Repp, 1983). Data about variable-rate speech perception (Repp, 1980, 1983) have been simulated using a STORE model in which the storage rate is adjusted by automatic gain control to speed up or slow down with the speech rate, leading to a stored STM pattern that is invariant across a wide range of rates (Boardman, Cohen, and Grossberg, 1993).

5. The Reduced Model: STORE 0

Before turning to the STORE 2 model, it is informative to ask whether the competence of STORE 1 can be achieved by a single-layer network. A single-layer system (STORE 0) can, in fact, encode an invariant working memory, but at a cost of losing the robustness to input timing that characterizes STORE 1.

Figure 2

In a single-layer STORE system, the STORE 1 positive feedback loop $F_1 \rightarrow F_2 \rightarrow F_1$ (Figure 1a) is replaced with direct $F_1 \rightarrow F_1$ positive feedback (Figure 2). This is a natural simplification, since the STORE 1 variable y_k records and feeds back prior values of x_k. Equation (4) describes STM dynamics of the one-layer system:

STORE 0: F_1 Working Memory

$$\frac{dx_i}{dt} = (AI_i + x_i - x_i x)I, \tag{4}$$

where I_i satisfies (1).

Figure 3

Figure 3a shows that, like STORE 1, STORE 0 can exhibit recency ($A = 1.3$), bowing ($A = 0.3$), and primacy ($A = 0.04$) gradients. Intuitively, parameter A is an index of the strength of the current input I_i relative to the positive feedback term x_i. Large A enhances the influence of the current input I_i relative to the STM representation $x_1, \ldots, x_{i-1}$ of past inputs, and so produces a recency gradient. Invariance is illustrated by the relative STM activities x_k/x_{k+1}, which remain constant through time as new inputs are added. Figure 3 also illustrates the normalization property; namely, the total F_1 STM activity:

$$S_i \equiv \sum_{k=1}^{i} x_k(t_i) \tag{5}$$

increases toward a constant asymptotic value S as the number of items stored in working memory increases. For both STORE 1 and STORE 0,

$$S = .5[1 + (1 + 4A)^{1/2}]. \tag{6}$$

Figure 3b illustrates that, unlike STORE 1, STORE 0 activity patterns are sensitive to input timing variations. In Figure 3a, where $\alpha_i = \beta_i = 0.75$, STM bows at position 4 when $A = 0.3$. In Figure 3b, where A also equals 0.3, bowing occurs later (position 7) when $\alpha_i = 0.3$; and earlier (position 2) when $\alpha_i = 1.2$. This property occurs in STORE 0 because STM values x_k $(k < i)$ decay toward 0 when input I_i is on for a long interval. Thus temporal storage in STORE 0 requires that the duration α_i of the input be short enough so that STM of previous items cannot reach a zero steady state. Shorter input durations (smaller α_i) give less weight to recent inputs, leading to a longer primacy gradient, while longer input durations (larger α_i) enhance the recency gradient. The length of the interstimulus interval (β_i) has no effect on the STORE 0 activity pattern, due to the gating term I in (4) that holds x_i constant when no input is present. Thus STORE 0 is an adequate working memory insofar as input preprocessing guarantees approximately equal input durations and intensities.

6. Control of STM Gradients: STORE 2

STORE 1 is perhaps the simplest neural model that is capable of invariant encoding and recall of temporal sequences in real time. However, with just one free parameter (A), STM gradients tend to be steep. Addition of another term (and parameter) to the model provides a new degree of freedom that brings greater flexibility to applications and cognitive modeling.

STORE 1 can be augmented in a variety of ways. One natural way is to include a working memory decay term ($-Bx_i$) to the description of the activations x_i at F_1; namely,

STORE 2: F_1 Working Memory

$$\frac{dx_i}{dt} = (AI_i + y_i - x_i x - Bx_i)I. \tag{7}$$

Equations (1), (3), and (7) constitute the STORE 2 model, which retains the same two-layer geometry as STORE 1 (Figure 1a) and reduces to STORE 1 when $B = 0$. In that case, primacy, and veridical recall, occur for small A, which gives a current input I_i less weight than past items, whose presentation order is retained in the F_2 values $y_1, \ldots, y_{i-1}$.

Figure 4

The decay term $-Bx_i$ modulates the steep STORE 1 activation gradients. Figure 4 shows the results of STORE 2 simulations that vary both the input strength parameter A and the STM decay parameter B. Each rectangle shows the evolving steady-state F_1 STM values $(x_1 \ldots x_7)$ as a sequence of inputs $I_1, \ldots, I_7$ is presented. For comparison, all activations $x_k(t_i)$ represented by the bar charts have been normalized by the total activity $(x(t_7))$, after the final input. From the left column to the right column, the STM decay parameter B is seen to "smooth out" the steep primacy gradient that often occurs in STORE 1. The additional degree of freedom in STORE 2 thus allows control of the shape of primacy, bowing, and recency curves, to keep STM values in a useable range, in particular above the noise level that may exist in real systems. We will now mathematically analyse STORE 2 dynamics as a function of the two free parameters A and B.

During presentation of the i^{th} input to a STORE 2 system, when $t_i - \alpha_i < t < t_i$, $I_i = 1$ and $y_i = 0$. Therefore

$$\frac{dx_i}{dt} = A - x_i x - Bx_i, \tag{8}$$

so

$$x_i \to \frac{A}{x+B}. \tag{9}$$

For $k < i$, $I_k = 0$ and $y_k \cong x_k(t_{i-1})$ during this interval (Figure 1b). Therefore

$$\frac{dx_k}{dt} \cong x_k(t_{i-1}) - x_k x - Bx_k, \tag{10}$$

so

$$x_k \to \frac{x_k(t_{i-1})}{x+B}. \tag{11}$$

By (11), the prior working memory pattern $(x_1 \ldots x_{i-1})$ is scaled by the common factor $(x+B)^{-1}$ when input I_i is being stored. Therefore relative activations are preserved, and STORE 2 satisfies the Invariance Principle. Note that storage of a new input I_i causes a net amplification of the prior pattern $(x_1(t_{i-1}) \ldots x_{i-1}(t_{i-1}))$ if and only if

$$x(t_i) + B \equiv S_i + B < 1, \tag{12}$$

by (5) and (11).

Equations for total STM activity at F_1 and F_2 are obtained by summing equations (3) and (7). Thus, setting $y \equiv \sum_k y_k$,

$$\frac{dx}{dt} = (A + y - x^2 - Bx)I \tag{13}$$

and

$$\frac{dy}{dt} = (x - y)I^c. \tag{14}$$

By design, $y \to x(t_{i-1})$ in the interval $[t_{i-1}, t_{i-1} + \beta_{i-1}]$ between input I_{i-1} and input I_i (Figure 1b), and y remains constant in the next interval $(t_i - \alpha_i, t_i)$ when input I_i is presented. Thus, by (5) and (14), $y(t_i) \cong x(t_{i-1}) \equiv S_{i-1}$; thus by (13),

$$A + S_{i-1} - S_i^2 - BS_i \cong 0. \tag{15}$$

Solving (15) then implies that the total F_1 activity $x(t_i) \equiv S_i$ is given by the iteration formula:

$$S_i = 0.5[-B + (B^2 + 4(A + S_{i-1}))^{1/2}], \tag{16}$$

where $S_0 \equiv 0$. Thus by (16), $S_1 > S_0$. Comparison of (16) evaluated at S_i and at S_{i+1} shows, by induction, that

$$S_1 < S_2 < \ldots < S_i < \ldots \tag{17}$$

at all times.

Equations (16) and (17) can now be used to calculate the position at which the pattern $(x_1, x_2, \ldots, x_n)$ may bow. STORE 2 exhibits a primacy gradient so long as

$$x_{i-1}(t_i) > x_i(t_i). \tag{18}$$

By (9) and (11), this occurs iff

$$x_{i-1}(t_{i-1}) > A. \tag{19}$$

Thus, by (9) and (19), bowing occurs at the first position $j = J$ at which

$$x_j(t_j) \cong \frac{A}{S_j + B} \leq A. \tag{20}$$

In addition, by (9), (17), and (20),

$$x_i(t_i) \cong \frac{A}{S_i + B} < A \tag{21}$$

for $i > J$, since total F_1 activity S_i grows monotonically as new inputs arrive, by (17). By (11) and (21), for all $i > J$,

$$x_{i-1}(t_i) < x_i(t_i). \tag{22}$$

In particular, if $B \geq 1$ in (20), then $J = 1$ and a recency gradient occurs. By (20), for $0 \leq B < 1$, bowing occurs at the first position $j = J$ where

$$S_j \geq 1 - B. \tag{23}$$

Figure 5

7. Repeated Input Items: STORE 3

When order is encoded in STM activation levels and when, as in STORE 1 or STORE 2, each item is represented by just one node, repeated items in an input stream pose a problem. Namely, repeated items could increase the activation level of the corresponding node in such a way that the order information encoded by relative activations is lost. To solve this problem, STORE 3 automatically creates new internal representations when an input item is repeated. As in Figure 5, a preprocessor at level F_0 represents repeated items in spatially separate channels. Both repeated and non-repeated items then enter level F_1 as spatially separate inputs. In this way, a STORE 3 network can be viewed as a 2-D array of *items* × *repeats*. Two methods for spatially separating repeated items in level F_0 are proposed here. The first uses inhibitory feedback from the STORE F_2 level to a winner-take-all competitive field F_0 (STORE 3 WTA) (Figure 5). The second uses a positional gradient shift at F_0 (STORE 3 PGS) that does not require feedback from the STORE network.

Figure 6

STORE 3 Winner-Take-All (WTA) Preprocessor

Figure 6 depicts the slice of the STORE 3 WTA network that encodes a single input I_σ to the item representation σ. A node that becomes active when item σ is recognized is connected, via n pathways, to a repeated-item preprocessor F_0^σ, which in turn feeds into the STORE 3 network. That is, each input I_σ sends excitatory signals $(r_1^\sigma I_\sigma, \ldots, r_n^\sigma I_\sigma)$ to an array of n nodes in a winner-take-all competitive field F_0^σ. Connection strengths r_j^σ are assumed to be fixed numbers that are randomly chosen in (0,1). The F_0^σ node J that receives the largest input becomes active, while activity at other nodes is inhibited. When activity at the winning node exceeds a threshold T, the corresponding J^{th} node in the STORE 3 field F_1^σ becomes active. After the input I_σ goes off, massive inhibition from the active J^{th} F_2^σ node prevents subsequent activation of the J^{th} F_0^σ node, until the entire STORE network is reset. Inhibition from F_2^σ allows repeated instances of input I_σ to excite distinct nodes in the winner-take-all network F_0^σ, which are chosen in order of decreasing size of the strengths r_j^σ.

Let σ_i denote the ith item representation to be activated in an event sequence. The STORE 3 WTA network encodes an arbitrary input sequence $I_{\sigma_1}, I_{\sigma_2}, \ldots, I_{\sigma_i}, \ldots$ as follows.

(A) For simplicity of notation, denote a fixed item representation σ_i by σ. Input $I_{\sigma_i} = I_\sigma$ fans out with randomly varying connection strengths r_j^σ to n nodes in the winner-take-all network F_0^σ during the interval $(t_i - \alpha_i, t_i)$.

(B) The F_0^σ node (J) with the largest weighted input $(r_J^\sigma I_\sigma)$ suppresses activity at the other nodes in F_0^σ.

(C) When activity (w_J^σ) of the winning node exceeds a threshold (T), output from the J^{th} F_0^σ node excites the J^{th} node of the STORE 3 layer F_1^σ.

(D) After input I_{σ_i} shuts off $(t_i \leq t \leq t_i + \beta_i)$, activity (y_j^σ) of each F_2^σ node delivers positive feedback to the corresponding F_1^σ node (x_j^σ) and a large inhibitory signal $-Ey_j^\sigma$ to the corresponding F_0^σ node (w_j^σ). In this way, each newly active F_2^σ node inhibits subsequent activation of the corresponding node in F_0^σ by repeats of the item σ.

(E) If input I_σ is repeated, a different F_0^σ node this becomes active. STORE 3 hereby treats repeated instances of a given input as if they were distinct inputs.

The dimensionless STORE 3 WTA network is characterized by equations (24)–(26). Table 1 describes STORE 3 parameters.

Table 1

STORE 3 WTA: F_0^σ Preprocessor

$$\frac{dw_j^\sigma}{dt} = C(-Dw_j^\sigma + (1 - w_j^\sigma)[f(w_j^\sigma) + r_j^\sigma I_\sigma] - w_j^\sigma[\sum_{k \neq j} f(w_k^\sigma) + Ey_j^\sigma]), \tag{24}$$

where $I_\sigma(t) = 1$ at times t when item σ is being presented, $I_\sigma = 0$ otherwise; $\sigma = 1, \ldots \sum$; $j = 1 \ldots n$; and $f(w) \equiv Fw^2$. See Grossberg (1973, 1982) for an analysis of the dynamics of such shunting on-center off-surround networks.

STORE 3 WTA: F_1^σ Working Memory

$$\frac{dx_j^\sigma}{dt} = \left(A[w_j^\sigma - T]^+ + y_j^\sigma - x_j^\sigma x - Bx_j^\sigma\right) I, \tag{25}$$

where $x \equiv \sum_\sigma \sum_j x_j^\sigma$, $I \equiv \sum_\sigma I_\sigma$, and $[z]^+ = \max(z, 0)$.

STORE 3 WTA: F_2^σ Stored Memory

$$\frac{dy_j^\sigma}{dt} = \left(x_j^\sigma - y_j^\sigma\right) I^c, \tag{26}$$

where $I^c \equiv 1 - I$.

Figure 7

Figure 7 summarizes a computer simulation of the winner-take-all preprocessor of the STORE 3 WTA layer F_0^σ. In Figure 7a, an input fans out with varying connection strengths to seven nodes in the winner-take-all network. Bar heights show evolving F_0^σ activities w_j^σ during a brief interval ($0 \leq t \leq 0.06$). The winner-take-all dynamics enhance F_0^σ activity at the node ($J = 5$) with maximum r_j^σ, and suppresses activity at other nodes ($j \neq 5$). Only the winning node exceeds the threshold (T) for sending a signal to F_1^σ (equation 25). Figure 7b shows the results of seven repeats of input item I_σ. Each instance activates a different F_0^σ node, leading to spatially separated activations in layer F_1^σ. Figure 8 illustrates STORE 3 working memory responses to various input sequences that include repeated items. In each case, F_1 activity encodes the correct input order, given a small value of parameter A to ensure that a primacy gradient unfolds.

Figure 8

STORE 3 Position Gradient Shift (PGS) Preprocessor

A second method of spatially separating repeated input items into different channels uses feedforward excitatory and inhibitory positional gradients to convert repeated inputs into changing locations in a spatial map. One such map, called a Position-Threshold-Slope (PTS) Shift map, was introduced by Grossberg and Kuperstein (1989) to transform different input intensities into different spatial locations. Another map, called a Difference-of-Difference-of-Gaussians (DODOG), was introduced by Gaudiano and Grossberg (1991) to convert different ratios of two input intensities into different spatial locations. Either map could be used herein as a preprocessor. If successive presentations of the same item are stored, then the total stored input increases with successive presentations and could be used as the input to a PTS Shift map. If each item input is broken into an excitatory and inhibitory input pathway and successive item presentations are stored in the inhibitory pathway, then the ratio of inputs in the two pathways changes with successive presentations and could be used as the input to a DODOG map.

The preprocessor that is described below is a variant of these models that realizes the desired mapping in a simple way. It is called a Position Gradient Shift (PGS) map. The PGS preprocessor includes inhibitory connections within the F_0^σ field, so inhibition does not need to feed back from F_2^σ, as in the STORE 3 WTA variant. Each input channel σ fans out via both excitatory and inhibitory connections, whose strengths fall off with distance, to a winner-take-all field F_0^σ. In each channel an inhibitory interneuron's activation Λ_σ grows with each repeat of input σ. The growing inhibitory gradient allows a different node in F_0^σ to become active with each repetition of I_σ. As with the WTA preprocessor, each F_0^σ node is connected to the STORE 3 level F_1, and each input event activates a different node in working memory.

Figure 9

Figure 9 shows the components of a positional gradient shift repeated item preprocessor. An transient cell activity Θ_σ converts a sustained input $I_{\sigma_i} = I_\sigma$ of duration α_i into a pulse of short fixed duration Δt via an inhibitory interneuron that shuts Θ_σ off after a brief time delay. These pulses feed into an integrator cell whose activity Λ_σ steps up with each transient pulse Θ_σ.

Figure 10

Figure 10 shows slice σ of the STORE 3 PGS network. Input $I_{\sigma_i} = I_\sigma$ both directly excites each F_0^σ node; and indirectly inhibits each node, via the integrator cell Λ_σ. Input I_σ excites F_0^σ nodes via signals whose size $[I_\sigma - \eta_+ j]^+$ decreases linearly with distance away from the excitatory I_σ input node. Similarly the size of inhibitory signals $[\Lambda_\sigma - \eta_- j]^+$ from the integrator cell to F_0^σ nodes decreases linearly with distance. It is assumed that the strength of the excitatory connections decreases more slowly than that of the inhibitory connections, moving from the F_0^σ cell $j = 1$ toward the cell $j = n$; that is, $\eta_- > \eta_+$. The combined effect of the excitatory input gradient from I_σ and the growing inhibitory gradient from Λ_σ is to shift by one node the locus of maximal F_0^σ activation with each repeat of item σ. In this manner, repeated inputs are spatially separated before their order is encoded in the STORE network, without using any feedback from the STORE network levels F_1 or F_2.

Equations (27)–(31), along with the F_1 equation (25) and the F_2 equation (26), characterize the STORE 3 PGS system. Parameters are given in Table 2.

Table 2

STORE 3 PGS: F_0^σ Preprocessor

Sustained Input

$$I_\sigma = \begin{cases} 1 & \text{for } t_i - \alpha_i < t < t_i, \text{ when } \sigma_i = \sigma \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

Transient Node

$$\Theta_\sigma(t) = \begin{cases} 1 & \text{for } \ t_i - \alpha_i < t < t_i - \alpha_i + \Delta t \\ 0 & \text{otherwise} \end{cases} \tag{28}$$

Inhibitory Integrator Node

$$\frac{d\Lambda_\sigma}{dt} = \Theta_\sigma \tag{29}$$

Excitatory Gradient Signals to F_0^σ

$$\phi_j^+(I_\sigma) = [I_\sigma - \eta_+ j]^+ \tag{30}$$

Inhibitory Gradient Signals to F_0^σ

$$\phi_j^-(\Lambda_\sigma) = [\Lambda_\sigma - \eta_- j]^+, \tag{31}$$

where $\eta_- > \eta_+ > 0$; $j = 1, \ldots, n$; and $[\lambda]^+ = \max(\lambda, 0)$.

F_0^σ **Winner-Take-All**

$$\frac{dw_j^\sigma}{dt} = C\Big[-Dw_j^\sigma + (I_\sigma - w_j^\sigma)[f(w_j^\sigma) + \phi_j^+(I_\sigma)] - w_j^\sigma\Big[\sum_{k \neq j} f(w_k^\sigma) + E\phi_j^-(\Lambda_\sigma)\Big]\Big]. \tag{32}$$

Figure 11

Figure 11 shows how the STORE 3 PGS model records repeated items in working memory. In Figure 11a, the fifth F_0^σ node (w_5^σ) receives the greatest combined input, $[I_\sigma - 5\eta_+]^+ - [\Lambda_\sigma - 5\eta_-]^+$ when σ is repeated for the fifth time. It therefore wins the competition and suppresses activity of the other F_0^σ nodes. The fifth node of F_1^σ then records in working memory the fifth instance of item σ. Figure 11b shows the evolving storage of seven repeats of item σ in working memory. Repeated items are seen to be processed into spatially separate channels prior to entering the STORE 3 network, where their order is subsequently encoded.

8. Alternative STORE Systems

The STORE idea of using two gated layers to create a working memory that invariantly records item and order information can be implemented in many ways. Three such systems are discussed below to illustrate variations on this general design theme. The first system is:

STORE 2A: F_1 **Working Memory**

$$\frac{dx_i}{dt} = \Big(x_i F + y_i + I_i\Big) I \tag{33}$$

along with equations (1) and (3). In (33),

$$F = A + By - x \tag{34}$$

where $A > 0$, $0 < B < 1$, $x = \sum_i x_i$, and $y = \sum_i y_i$. In (33), both excitatory and inhibitory nonspecific feedback are allowed to modulate each x_i, with inhibitory feedback stronger. Figure 12 demonstrates that bowing can occur at any position, with gradual STM primacy and recency gradients. Input duration was varied randomly from $\alpha_i = 10$ to $\alpha_i = 40$ without affecting order of storage.

Figure 12

The second system symmetrizes the feedback between F_1 and F_2:

STORE 2B: F_1 **Working Memory**

$$\frac{dx_i}{dt} = (x_i F + y_i + I_i) I \tag{35}$$

STORE 2B: F_2 **Stored Memory**

$$\frac{dy_i}{dt} = \Big(y_i G + x_i\Big) I^c, \tag{36}$$

where F is defined as in (34) and

$$G = A + Bx - y. \tag{37}$$

Figure 13 shows STORE 2B simulations with parameters set for primacy. Inputs were entered singly; two at a time; 2, 1, 3, 2 at a time; then in a pattern of 3, 1, 3, 1. This demonstrates that invariance is preserved even if inputs do not arrive sequentially.

Figure 13

The third system uses:

STORE 2C: F_1 Working Memory

$$\frac{dx_i}{dt} = (AI_i + f(y)y_i - x_i)I, \tag{38}$$

along with equations (1) and (3). In the other STORE 2 models, nonspecific inhibitory feedback $(-x)$ increases its effect on x_i as more items are stored. In STORE 2C, there is no nonspecific inhibitory feedback x. It is replaced by nonspecific excitatory feedback $f(y)$ that decreases its effect on y_i as more items are stored. Thus $f(y)$ in (38) is a positive decreasing function of total F_2 activity y, such as

$$f(y) = K - \epsilon y, \tag{39}$$

where $K > 1$ and $0 < \epsilon \leq 1$. The position of the bow in STORE 2C depends on where $f(y)$ becomes less than 1. Simulation results for STORE 2C are shown in Figure 14 where bowing at various positions is demonstrated.

Figure 14

9. Recognition and Prediction of Temporal Event Sequences

Invariant working memories are typically applied, in both biological and technological applications, as part of larger system architectures. The ability to stably learn to group sequences of real-time events is useful in applications to variable-rate speech perception, sensory-motor planning, and 3-D visual object recognition. In speech perception applications, such groupings include phonemic, syllabic, and word representations (Cohen and Grossberg, 1986; Grossberg, 1986). In sensory-motor planning, the groupings are often sequences of target position commands which describe spatial or motor representations of desired limb configurations (Grossberg and Kuperstein, 1989). In 3-D visual object recognition, the individual items represent individual views of an object (Bradski, Carpenter, and Grossberg, 1992). Grouped item sequences implicitly represent a 3-D object in terms of a stored sequence of 2-D views.

More generally, invariance properties of a STORE network enable them to be used as a processing substrate from which temporally evolving recognition codes, rules, or inferences may be learned. In particular, a STORE model can be used as the input level of a neural network categorizer or production system. A recently discovered family of Adaptive Resonance Theory networks, generically called ARTMAP (Carpenter and Grossberg, 1991, 1992; Carpenter, Grossberg, Markuzon, Reynolds, and Rosen, 1992; Carpenter, Grossberg, and Reynolds, 1991), is capable of supervised learning, categorization, and inference about arbitrary input vectors. In particular, ARTMAPs can learn arbitrary analog or binary mappings between learned categories of an input feature space (e.g., a STORE item and order code) to learned categories of an output feature space (e.g., predictions or names). A predictive error to the output feature space drives a bout of hypothesis testing to discover, focus attention upon, and learn about a more informative bundle of features in the input space. Using such bouts of hypothesis testing, ARTMAP architectures are capable of autonomously learning many-to-one and one-to-many mappings from input to output categories. A user can extract from these maps an algorithmic set of if-then rules at any stage of learning. ARTMAPs thus embody a type of self-organizing production system which sheds new light on how humans can realize rule-like behavior although their brains are not algorithmically structured in any traditional sense. These networks also embody heuristics which enable them to use predictive errors to match the degree of generalization of their learned categories, and the abstractness of their learned rules, to the demands of a particular input environment.

An architecture that combines ART and STORE networks is generically called an ARTSTORE system (Bradski, Carpenter, and Grossberg, 1992). Because a STORE model satisfies the Invariance Principle, an ARTSTORE system can selectively attend and learn those stored sequences of past events or actions that predict a desired outcome. Using these properties, ARTSTORE models provide a promising new approach to solving the subgoal planning problems that form a core part of human and animal problem solving in complex and rapidly changing environments.

REFERENCES

Atkinson, R.C. and Shiffrin, R.M. (1971). The control of short term memory. *Scientific American*, August, 82–90.

Baddeley, A.D. (1986). **Working memory**. Oxford: Clarendon Press.

Boardman, I. and Bullock, D. (1991). A neural network model of serial order recall from short-term memory. In **Proceedings of the international joint conference on neural networks**, Seattle, **II**, 879–884. Piscataway, NJ: IEEE Service Center.

Boardman, I., Cohen, M.A., and Grossberg, S. (1993). Variable rate working memories for phonetic categorization and invariant speech perception. In **Proceedings of the international joint conference on neural networks**, Portland, **III**, 2–5. Piscataway, NJ: IEEE Service Center.

Bradski, G., Carpenter, G.A. and Grossberg, S. (1991). Working memory networks for learning multiple groupings of temporally ordered events: Applications to 3-D visual object recognition, **Proceedings of the international joint conference on neural networks**, Seattle, **1**, 723–728. Piscataway, NJ: IEEE Service Center.

Bradski, G., Carpenter, G.A. and Grossberg, S. (1992). Working memory networks for learning temporal order with application to 3-D visual object recognition. *Neural Computation*, **4**, 270–286.

Carpenter, G.A. and Grossberg, S. (Eds.) (1991). **Pattern recognition by self-organizing neural networks**. Cambridge, MA: MIT Press.

Carpenter, G.A. and Grossberg, S. (1992). A self-organizing neural network for supervised learning, recognition, and prediction. *IEEE Communications*, **30**, 38–49.

Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., and Rosen, D.B. (1992). Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, **3**, 698–713.

Carpenter, G.A., Grossberg, S., and Reynolds, J.H. (1991). ARTMAP: Supervised real-time learning and classification of nonstationary data b a self-organizing neural network. *Neural Networks*, **4**, 565–588.

Cohen, M.A. and Grossberg, S. (1986). Neural dynamics of speech and language coding: Developmental programs, perceptual grouping, and competition for short term memory. *Human Neurobiology*, **5**, 1–22.

Elman, J.L. (1990). Finding structure in time. *Cognitive Science*, **14**, 179–211.

Gaudiano, P. and Grossberg, S. (1991). Vector associative maps: Unsupervised real-time error-based learning and control of movement trajectories. *Neural Networks*, **4**, 147–183.

Grossberg, S. (1970). Some networks that can learn, remember, and reproduce any number of complicated space-time patterns, II. *Studies in Applied Mathematics*, **49**, 135–166.

Grossberg, S. (1973). Contour enhancement, short term memory, and constancies in reverberating neural networks. *Studies in Applied Mathematics*, **52**, 217–257.

Grossberg, S. (1976). Adaptive pattern classification and universal recoding, I: Parallel development and coding of neural feature detectors. *Biological Cybernetics*, **23**, 121–134.

Grossberg, S. (1978a). A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans. In R. Rosen and F. Snell (Eds.), **Progress in theoretical biology, Volume 5**. New York: Academic Press, pp. 233–374.

Grossberg, S. (1978b). Behavioral contrast in short term memory: Serial binary memory models or parallel continuous memory models? *Journal of Mathematical Psychology*, **17**, 199–219.

Grossberg, S. (1982). **Studies of mind and brain: Neural principles of learning, perception, development, cognition, and motor control**. Boston: Reidel Press.

Grossberg, S. (1986). The adaptive self-organization of serial order in behavior: Speech, language, and motor control. In E.C. Schwab and H.C. Nusbaum (Eds.), **Pattern recognition by humans and machines, Volume 1: Speech perception**. New York: Academic Press, pp. 187–294. Reprinted in S. Grossberg (Ed.), **The adaptive brain, II: Vision, speech, language, and motor control**. Amsterdam: Elsevier/North-Holland, 1987, pp. 311–400.

Grossberg, S. and Kuperstein, M. (1989). **Neural dynamics of adaptive sensory-motor control: Expanded edition**. Elmsford, NY: Pergamon Press.

Grossberg, S. and Pepe, J. (1971). Spiking threshold and overarousal effects in serial learning. *Journal of Statistical Physics*, **3**, 95–125.

Grossberg, S. and Stone, G.O. (1986a). Neural dynamics of attention switching and temporal order information in short term memory. *Memory and Cognition*, **14**, 451–468.

Grossberg, S. and Stone, G.O. (1986b). Neural dynamics of word recognition and recall: Attentional priming, learning, and resonance. *Psychological Review*, **93**, 46–74.

Gutfreund, H. and Mezard, M. (1988). Processing of temporal sequences in neural networks. *Physiology Review Letters*, **61**, 235–238.

Guyon, I., Personnaz, L., Nadal, J.P. and Dreyfus, G. (1988). Storage retrieval of complex sequences in neural networks. *Physiology Review A*, **38**, 6365–6372.

Healy, A.F. (1975). Separating item from order information in short-term memory. *Journal of Verbal Learning and Verbal Behavior*, **13**, 644–655.

Jordan, M.I. (1986). Serial order: A parallel distributed processing approach. **Institute for Cognitive Science, Report 8604**. University of California, San Diego.

Lee, C. and Estes, W.K. (1981). Item and order information in short-term memory: Evidence for multilevel perturbation processes. *Journal of Experimental Psychology: Human Learning and Memory*, **1**, 149–169.

Miller, J.L. (1981). Effects of speaking rate on segmental distinctions. In P.D. Eimas and J.L. Miller (Eds.), **Perspectives on the study of speech**. Hillsdale, NJ: Lawrence Erlbaum Associates.

Miller, J.L. and Liberman, A.M. (1979). Some effects of later-occurring information on the perception of stop consonant and semivowel. *Perception and Psychophysics*, **25**(6), 457–465.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, **85**, 59–108.

Reeves, A. and Sperling, G. (1986). Attention gating in short-term visual memory. *Psychological Review*, **93**, 180–206.

Repp, B.H. (1980). A range-frequency effect on perception of silence in speech. **Haskins Laboratories Status Report on Speech Research, SR-61**, 151–165.

Repp, B.H. (1983). Bidirectional contrast effects in the perception of VC-CV sequences. *Perception and Psychophysics*, **33**, 147–155.

Repp, B., Liberman, A., Eccardt, T., and Pestsky, D. (1978). Perceptual integration of acoustic cues for stop fricative and affricate manner. *Journal of Experimental Psychology: Human Perception and Performance*, **4**(4), 621–637.

Schreter, Z. and Pfeifer, R. (1989). Short-term memory/long-term memory interactions in connectionist simulations of psychological experiments on list learning. In L. Personnaz and G. Dreyfus (Eds.), **Neural networks from models to applications**. Paris: I.D.S.E.T.

Seibert, M.C. (1991). Neural networks for machine vision: Learning three-dimensional object recognition. PhD Dissertation, Boston University.

Seibert, M.C. and Waxman, A.M. (1990a). Learning aspect graph representations from view sequences. In D.S. Touretzky (Ed.), **Advances in neural information processing systems 2**. San Mateo, CA: Morgan Kaufmann Publishing, pp. 258–265.

Seibert, M.C. and Waxman, A.M. (1990b). Learning aspect graph representations of 3D objects in a neural network. In M. Caudill (Ed.). **Proceedings of the international joint conference on neural networks**, Washington, DC, **2**, pp. 233–236. Hillsdale, NJ: Erlbaum.

Tartter, V.C., Kat, D., Samuel, A.G. and Repp, B.H. (1983). Perception of intervocalic stop consonants: the contributions of closure duration and formant transitions. *Journal of the Acoustical Society of America*, **73**(3), 715–725.

Wang, D. and Arbib, M.A. (1990). Complex temporal sequence learning based on short-term memory. *Proceedings of the IEEE*, **78**(9), 1536–1543.

FIGURE CAPTIONS

Figure 1. (a) Two-layer STORE 1 model. Layer F_1 is a competitive network whose variables x_k relax to steady state when an input is active in F_0. Level F_2 variables y_k track F_1 activity when inputs are off. In STORE 1 items are not repeated within a single working memory sequence. (b) Input timing. (c) An input sequence whose items enter in the order A, B, C can be stored in F_1 as a primacy, bowed, or recency gradient. The height of a line indicates the level of STM activity.

Figure 2. Single-layer STORE 0 model.

Figure 3. STORE 0 STM activity patterns at F_1 depend on the length of the input presentation interval $(t_i - \alpha_i, t_i)$. (a) Recency ($A = 1.3$), bowing ($A = 0.3$), and primacy ($A = 0.04$) gradients with the input presentation interval (α_i) held fixed. (b) Sensitivity to α_i with parameter A held fixed at 0.03. A shorter input interval ($\alpha_i = 0.3$) gives less weight to recent inputs, resulting in a stronger primacy gradients. A longer input interval ($\alpha_i = 1.2$) strengthens the recency gradient. STORE 0 exhibits invariance (constant x_k/x_{k+1}) and normalization (total activity S_i increasing towards an asymptotic value that is independent of the number of active nodes).

Figure 4. STORE 2. Steady state activations $(x_1, \ldots, x_7)$ normalized by total activity $x(t_7)$. The decay parameter B is seen to moderate the primacy gradient. Arrows indicate bow position.

Figure 5. STORE 3 WTA. Repeated items are filtered at F_0 into spatially separate channels and thus enter the STORE network as if they were separate inputs. An input I_σ activates one of n nodes in the F_0 layer of the σ^{th} "slice".

Figure 6. Slice σ of the STORE 3 WTA network: Repeated input items are separated into spatially distinct channels prior to encoding by STORE. Input I_σ fans out with randomly varying connection strengths r_j^σ into a winner-take-all field F_0^σ. Inhibition from F_2^σ to F_0^σ prevents subsequent activation of the j^{th} F_0^σ node. A repeat of input I_σ then causes another F_0^σ node to become active.

Figure 7. (a) F_0^σ chooses the node $J = 5$ with maximum path strength r_j^σ. F_0^σ reaches steady state rapidly compared to the input presentation time scale ($\alpha_i = \beta_i = 25$). (b) Seven repeats of item I_σ activate seven different F_0^σ nodes. A working memory activation pattern at F_1^σ can be used to learn and recall this sequence. Parameters are given in Table 1.

Figure 8. Response of STORE 3 WTA working memory to sequences with repeated items. Bar heights represent equilibrated activations x_j^σ in F_1, where input order is correctly encoded. Parameters are given in Table 1.

Figure 9. STORE 3 PGS integrator subcircuit.

Figure 10. STORE 3 PGS network.

Figure 11. STORE 3 PGS simulation. (a) Upon the fifth repetition of input I_σ, node $J = 5$ wins the competition at F_0^σ. (b) Increasing inhibition from the integrator node Λ_σ allows successive F_0^σ nodes $j = 1, \ldots, 7$ to become active as I_σ is presented seven times. STORE 3 records the seven repetitions in working memory.

Figure 12. STORE 2A. Bowing can occur in any position for this network. For each run, input durations were varied randomly from $\alpha_i = 10$ to $\alpha_i = 40$ without affecting order of storage.

Figure 13. STORE 2B. In this simulation, parameters were set to exhibit primacy over eight input presentations. Inputs were entered in different patterns: singly, doubly, and in patters of $2, 1, 3, 2$, and $3, 1, 3, 1$ as a demonstration that STORE networks can handle inputs in parallel if required. Simultaneous inputs are encoded with identical activation levels. Parameters in this system can also be set for arbitrary bow positions.

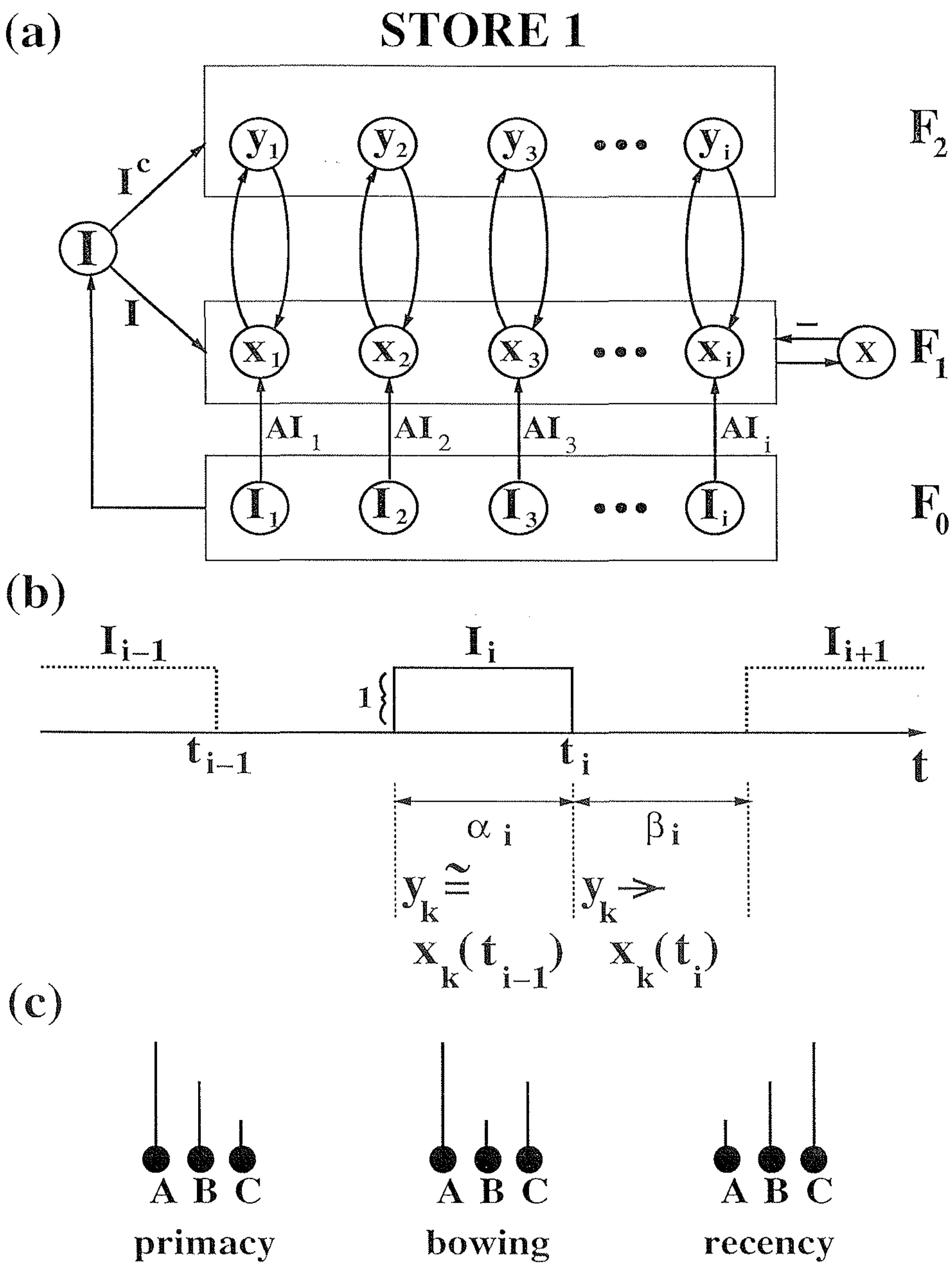
Figure 14. STORE 2C derived from algebraic constraints.

Parameter	Description
$A = 0.02$	Small, for primacy (Figure 3)
$B = 0.7$	Modulate F_1 gradient (STORE 2)
$C = 10$	$F_0 \rightarrow$ equilibrium before F_1 active
$D = 0.01$	Slow decay at F_0^σ
$E = 1000$	$>> 1$: for y_j^σ to quench w_j^σ
$F = 40$	Large: rapid choice at F_0^σ
$T = 0.5$	Prevent F_0^σ transients from activating F_1^σ
$0 < r_j^\sigma < 1$	Random coefficients;
	here, $r_1^\sigma > r_2^\sigma > \ldots > r_n^\sigma$
$n = 7$	Maximum number repetitions/item
$\alpha_i = 25$	Intra-input duration ($>> 1$)
$\beta_i = 25$	Inter-input duration ($>> 1$)

Table 1: STORE 3 WTA parameters.

Parameter	Description
$A = 0.02$	Small, for primacy
$B = 0.7$	Modulate F_1 gradient
$C = 10$	$F_0 \rightarrow$ equilibrium before F_1 active
$D = 0.01$	Slow decay at F_0^σ
$E = 8$	Inhibition weighting factor influences choice
$F = 40$	Large: rapid choice at F_0^σ
$T = 0.5$	Prevent F_0^σ transients from activating F_1^σ
$\Delta t = 0.1$	Input pulse duration
$\eta_+ = 0.05$	Excitatory signal falloff slope
$\eta_- = 0.1$	Inhibitory signal falloff slope ($> \eta_+$)
$n = 7$	Maximum number repetitions/item
$\alpha_i = 25$	Intra-input duration ($>> 1$)
$\beta_i = 25$	Inter-input duration ($>> 1$)

Table 2: STORE 3 PGS parameter summary.



(a)
STORE 1
F_2
y_1 y_2 y_3 ... y_i
I^c
I
I
F_1
x_1 x_2 x_3 ... x_i
–
X
AI_1 AI_2 AI_3 AI_i
F_0
I_1 I_2 I_3 ... I_i
(b)
I_{i-1}
I_i
I_{i+1}
1
t_{i-1}
t_i
t
α_i
β_i
$y_k \cong x_k(t_{i-1})$
$y_k \rightarrow x_k(t_i)$
(c)
A B C
primacy
A B C
bowing
A B C
recency

Figure 1

STORE 0

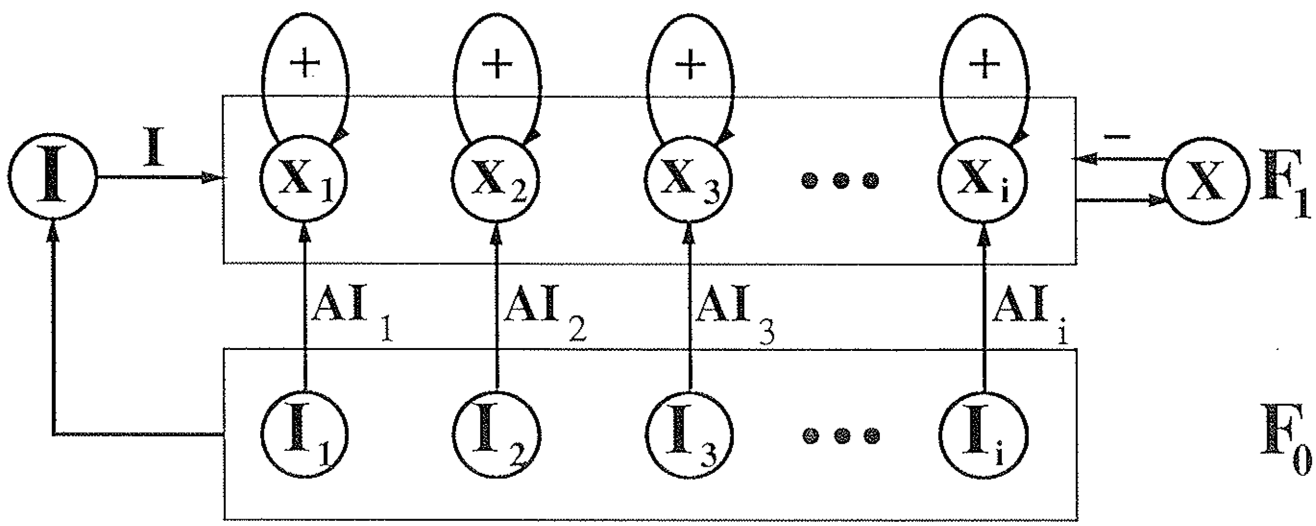


Figure 2

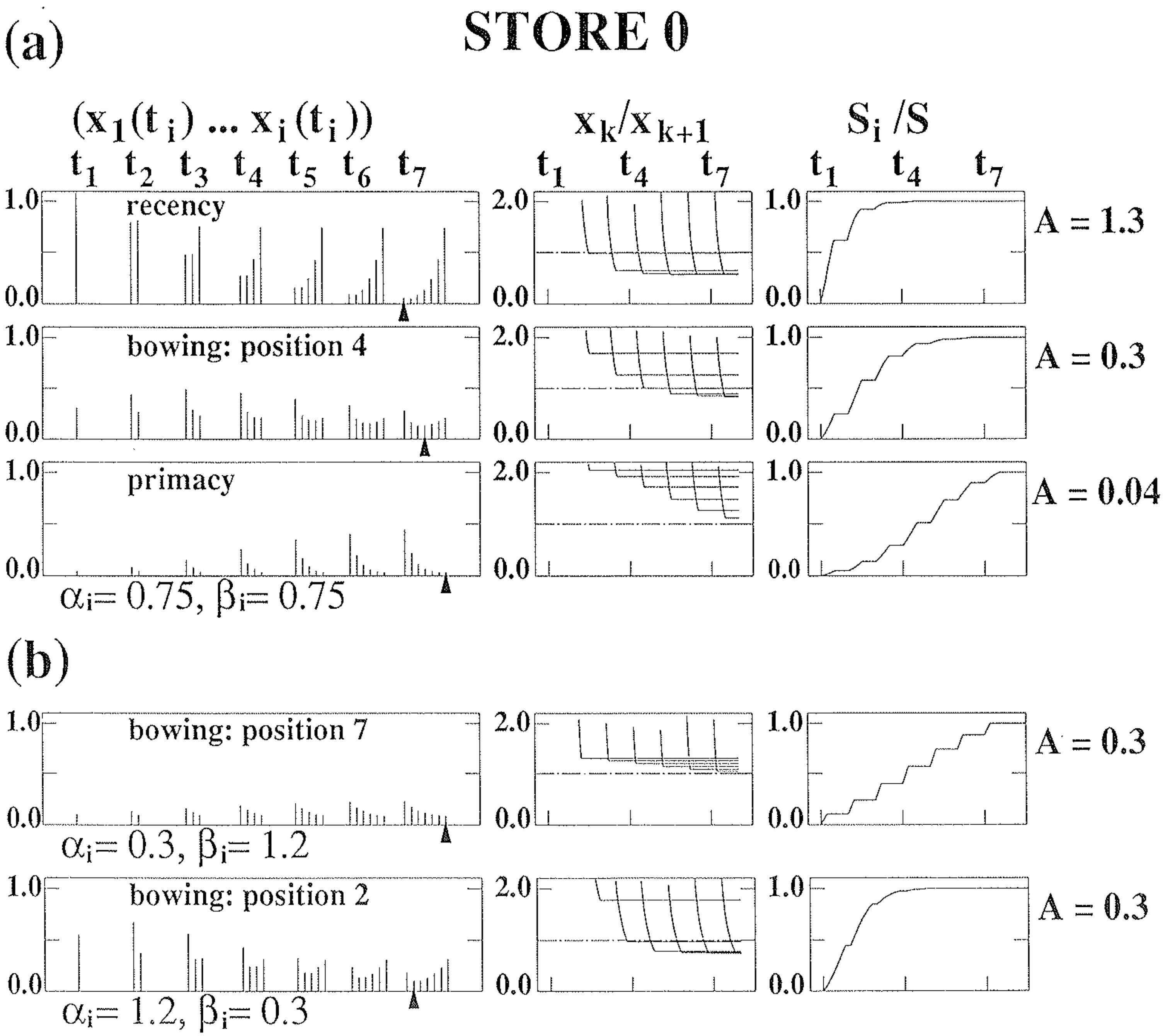

(a)
STORE 0
$(x_1(t_i) \ldots x_i(t_i))$
x_k/x_{k+1}
S_i/S
t_1 t_2 t_3 t_4 t_5 t_6 t_7
recency
A = 1.3
bowing: position 4
A = 0.3
primacy
A = 0.04
$\alpha_i = 0.75$, $\beta_i = 0.75$
(b)
bowing: position 7
A = 0.3
$\alpha_i = 0.3$, $\beta_i = 1.2$
bowing: position 2
A = 0.3
$\alpha_i = 1.2$, $\beta_i = 0.3$

Figure 3

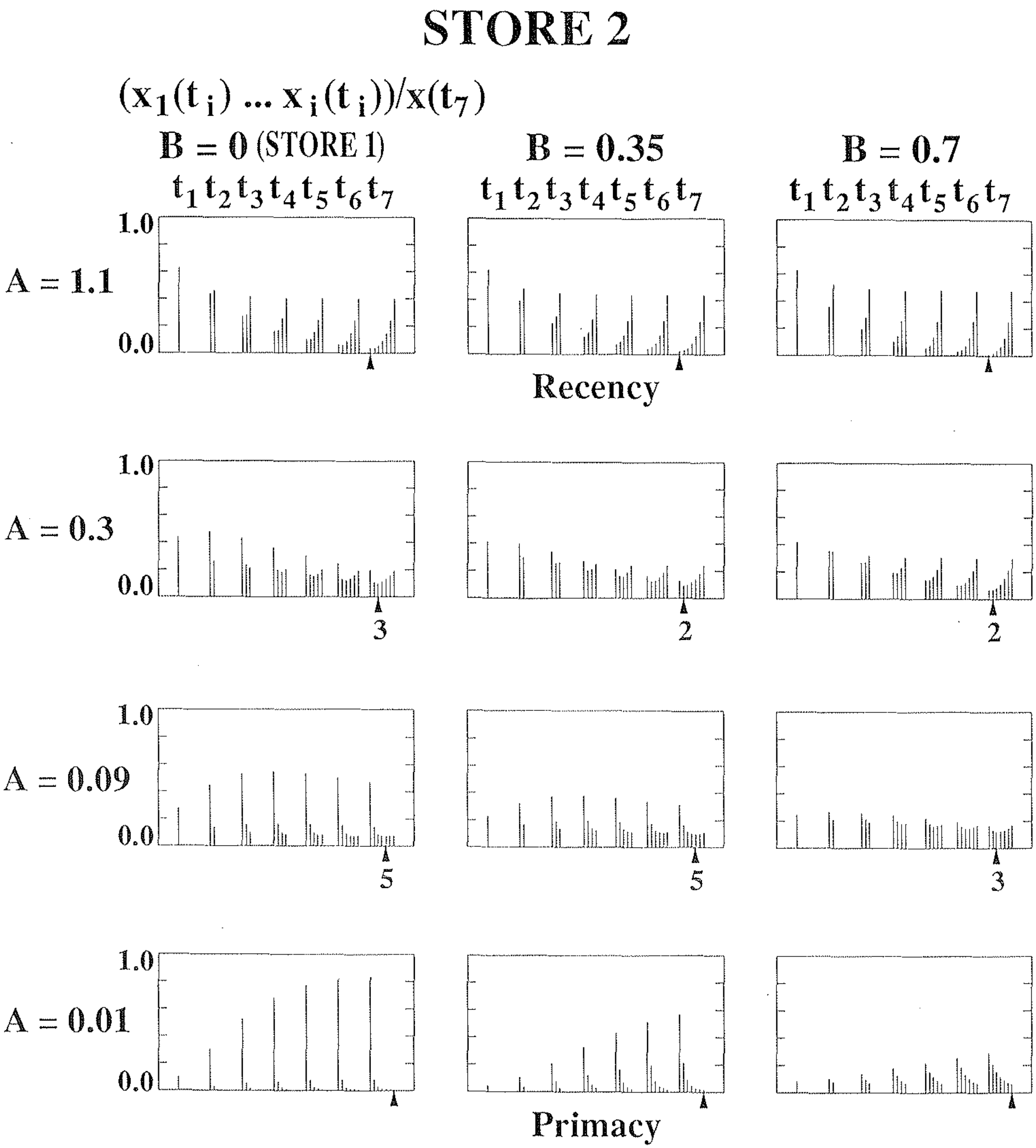

STORE 2
$(x_1(t_i) \ldots x_i(t_i))/x(t_7)$
B = 0 (STORE 1)
B = 0.35
B = 0.7
$t_1 t_2 t_3 t_4 t_5 t_6 t_7$
1.0
0.0
A = 1.1
A = 0.3
A = 0.09
A = 0.01
Recency
Primacy
3
2
2
5
5
3

Figure 4

STORE 3 WTA

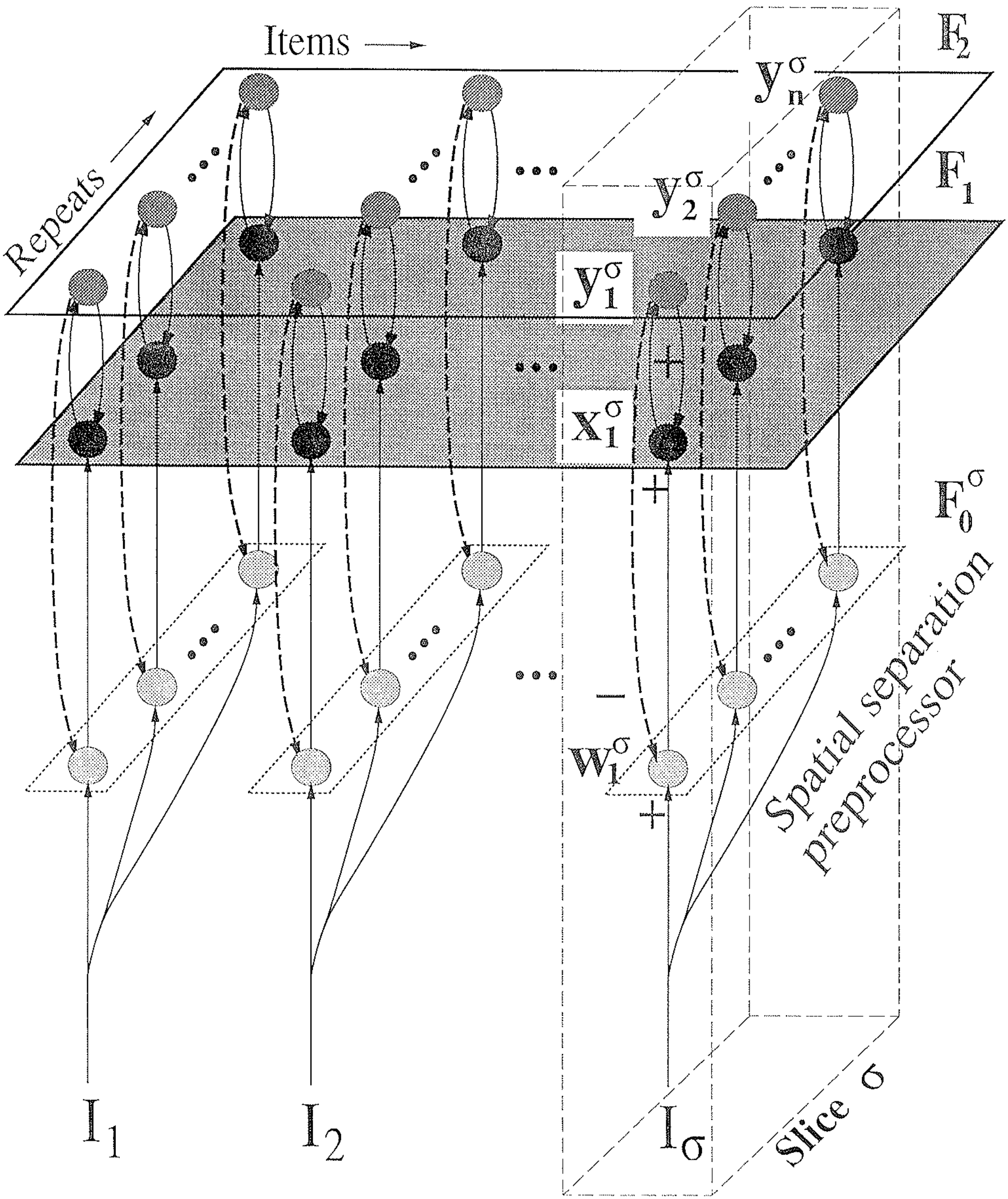


Figure 5

STORE 3 WTA

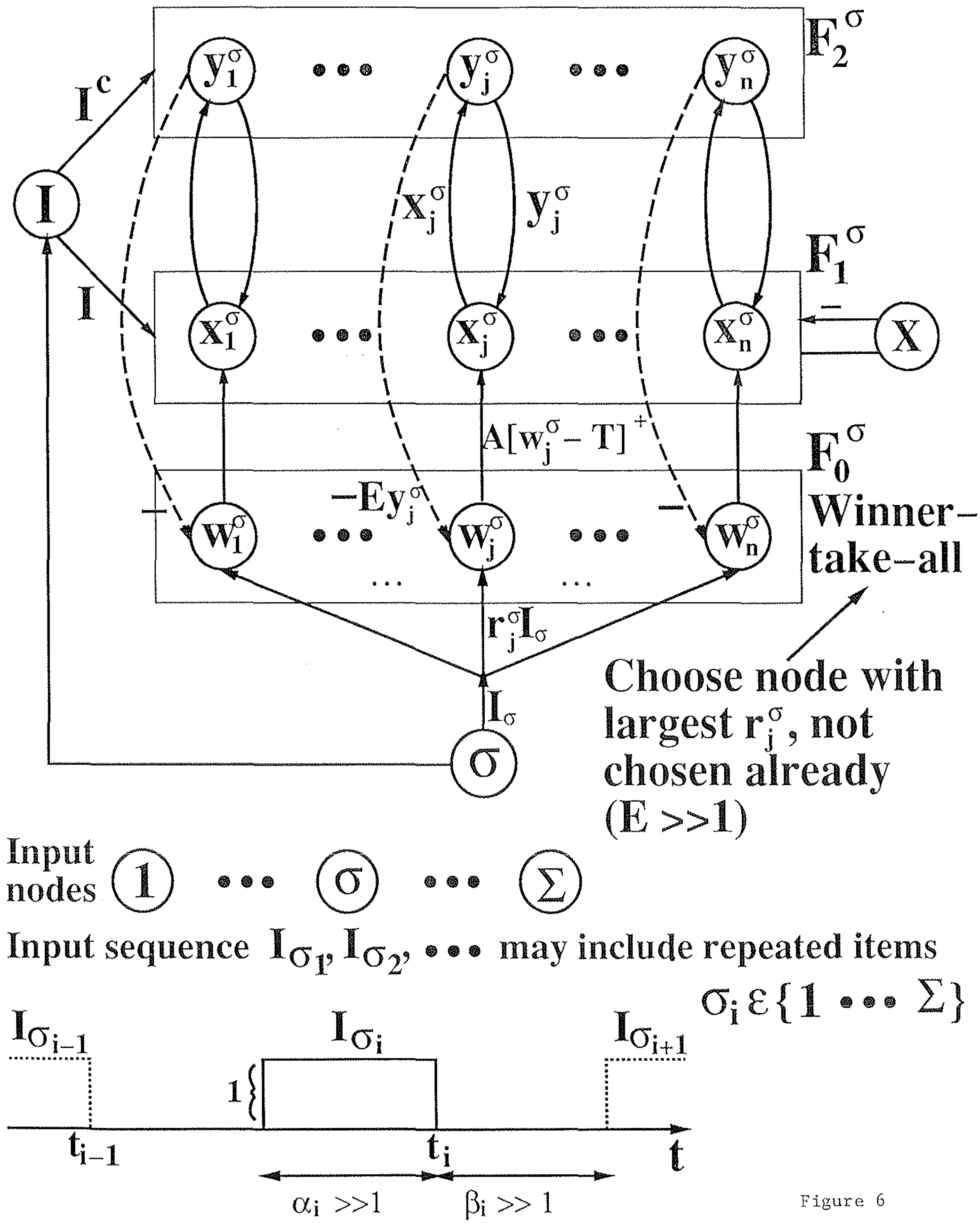
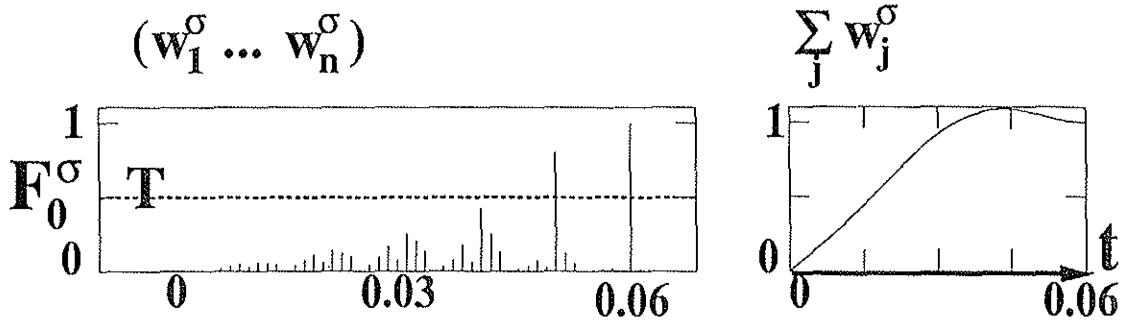


Figure 6

STORE 3 WTA

(a) Choice at winner–take–all field



(b) Seven repeats of input σ

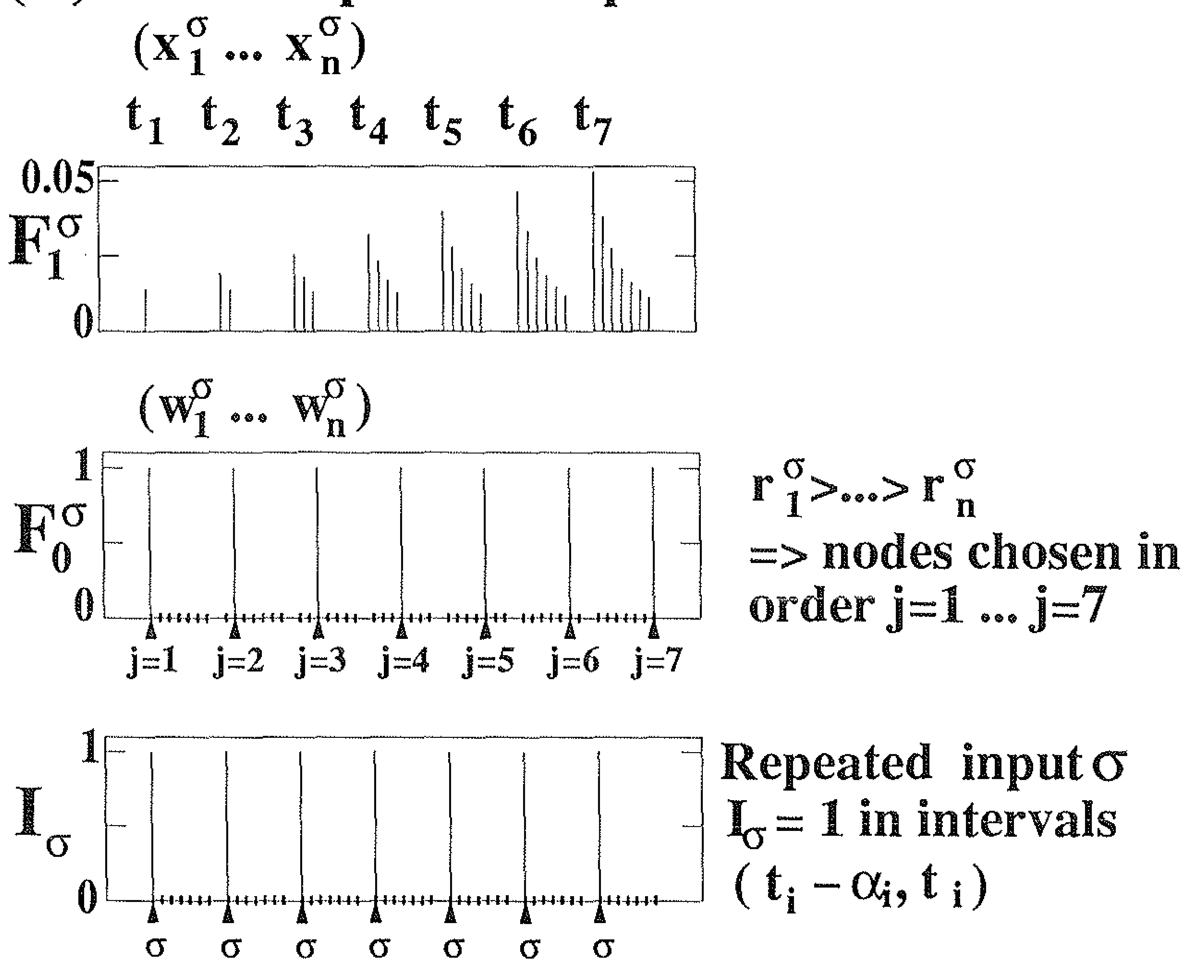
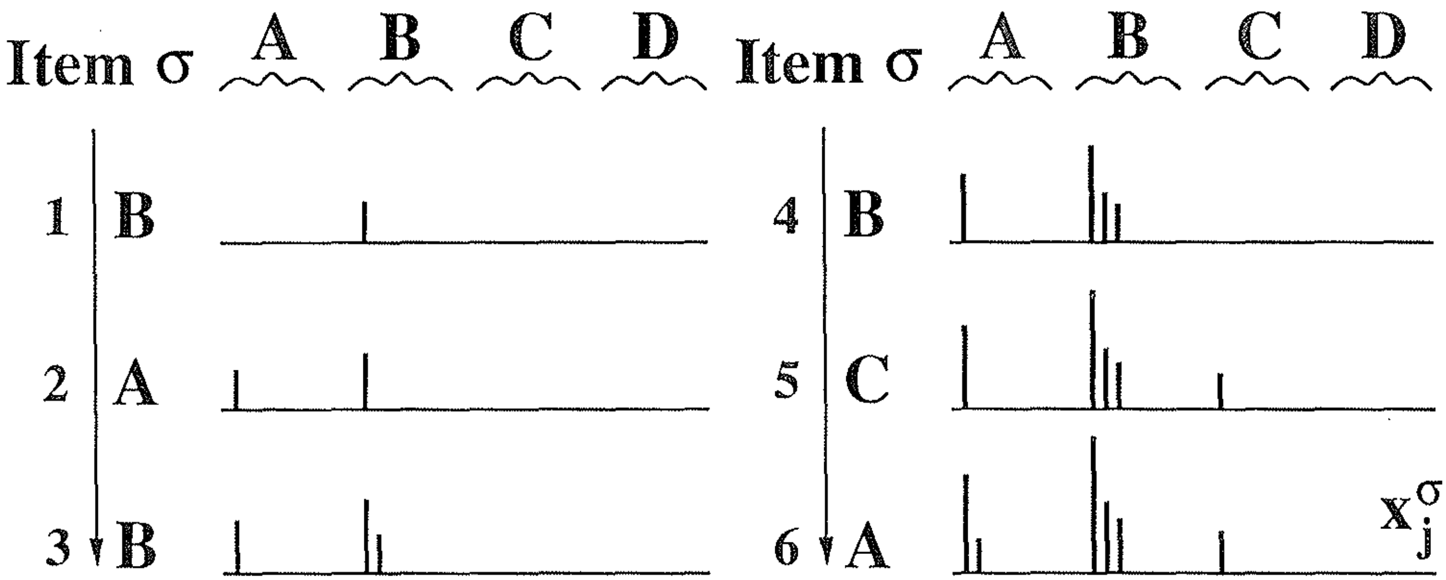


Figure 7

STORE 3

(a) Step-by-step response to BABBCA



(b) Final response to other sequences:

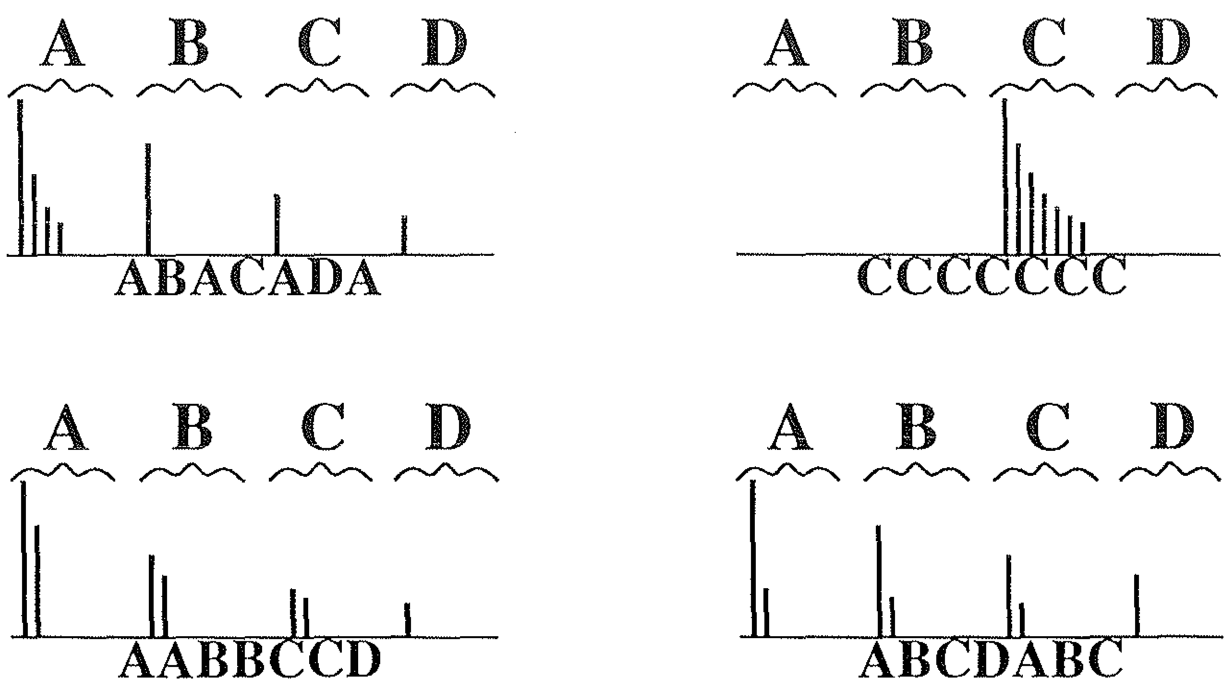


Figure 8

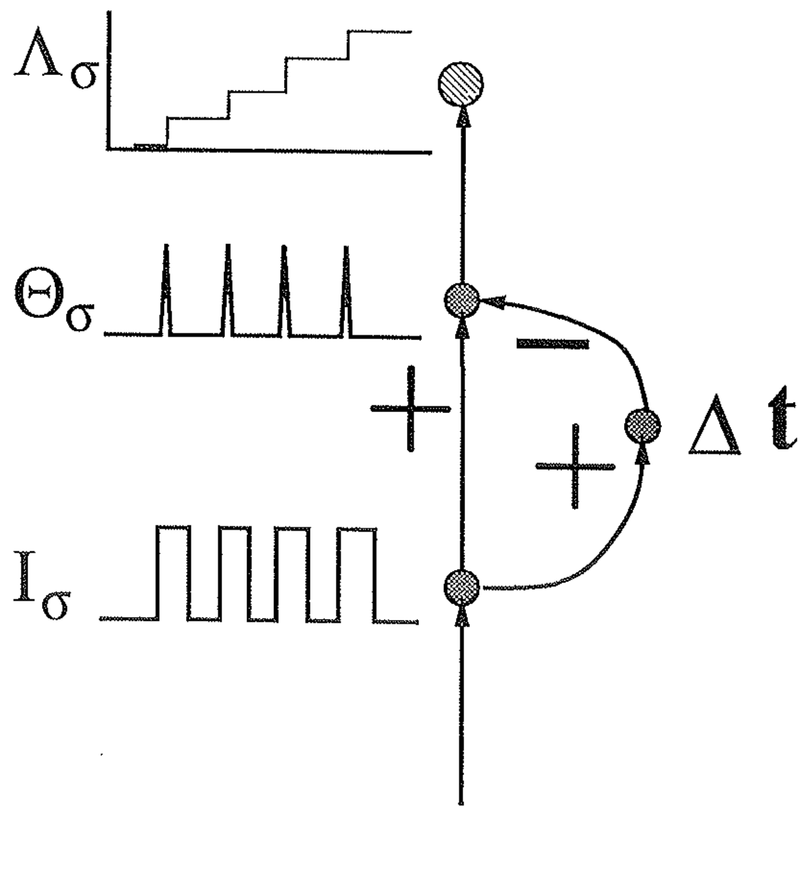


Figure 9

STORE 3 PGS

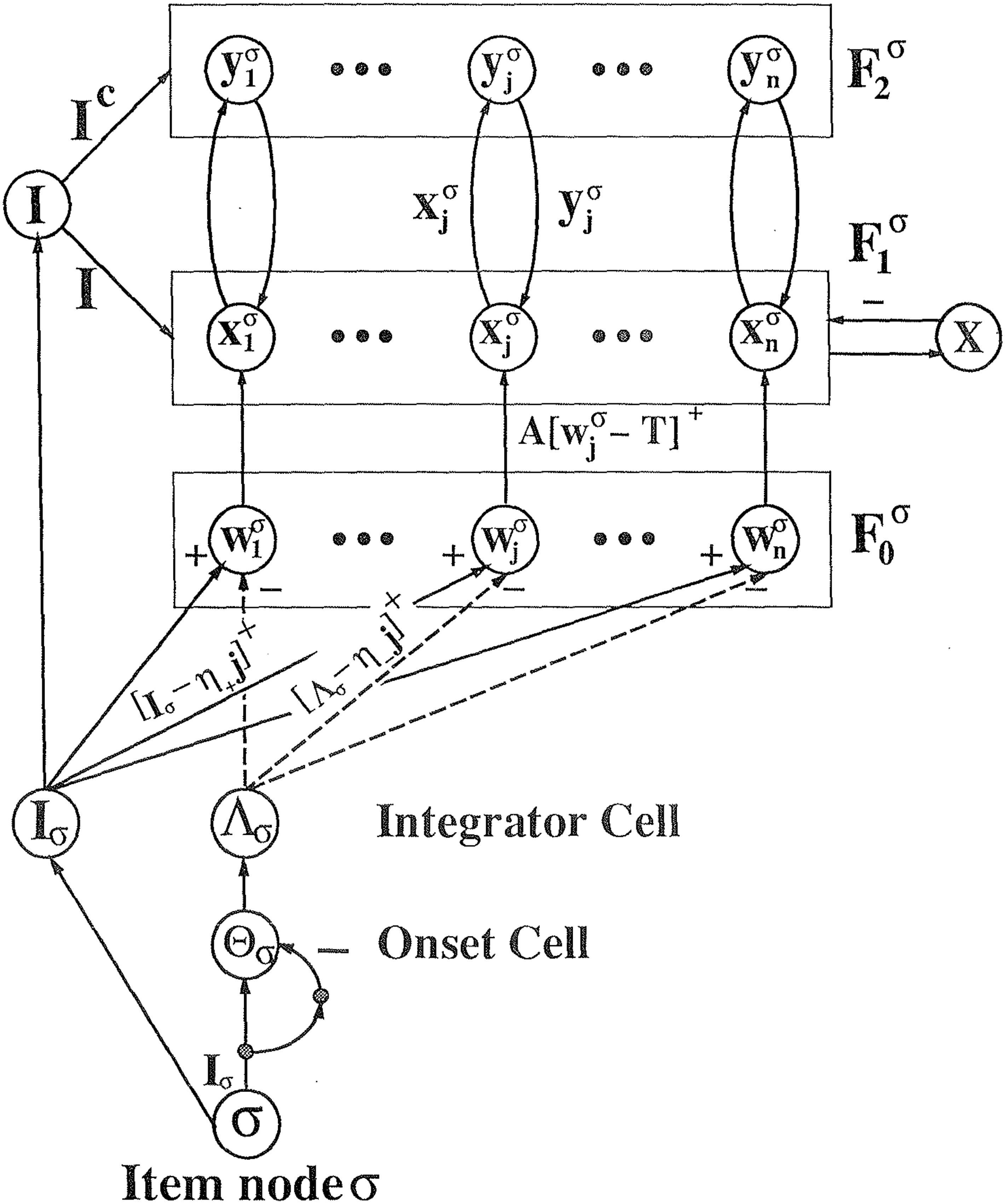


Figure 10

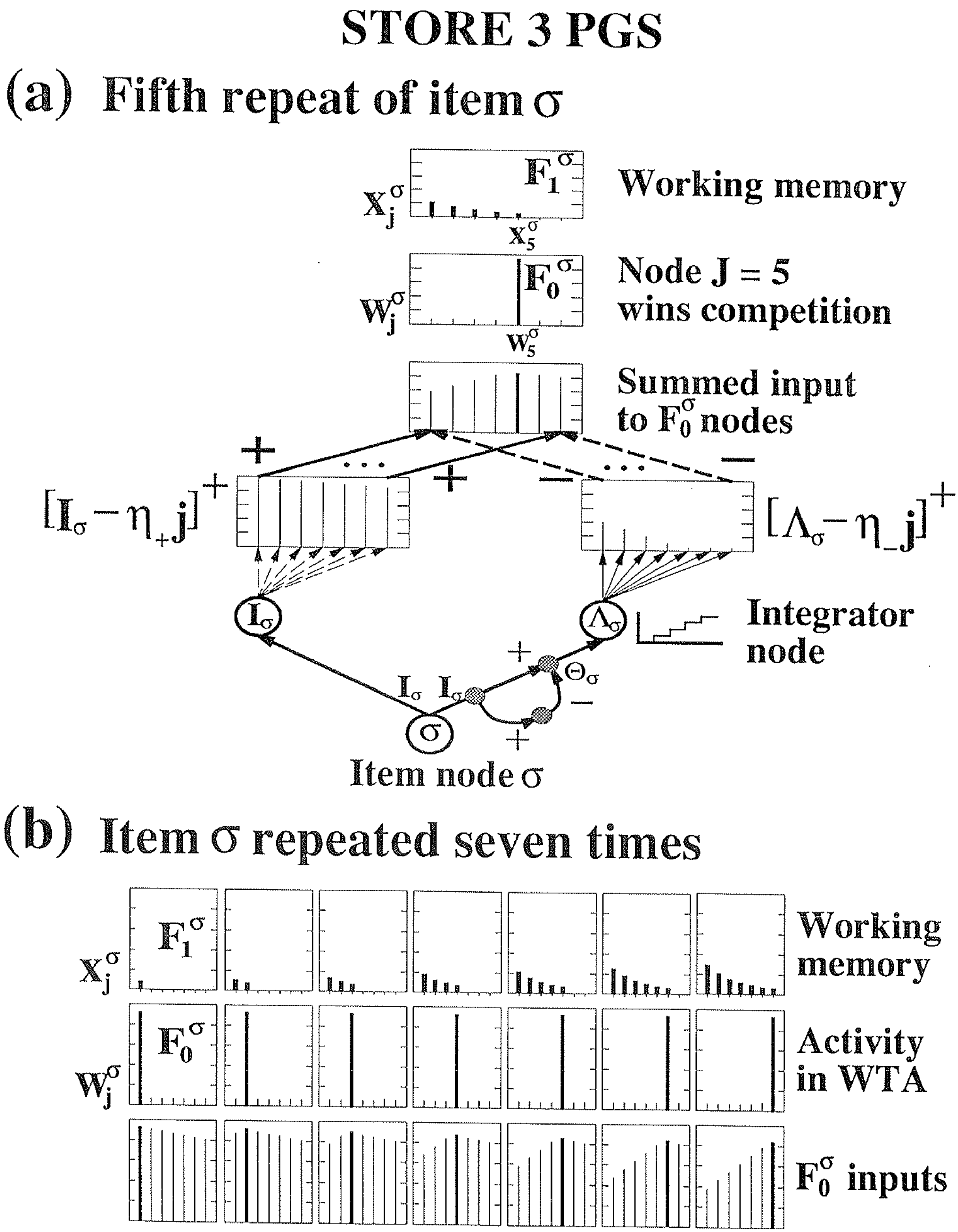

STORE 3 PGS
(a) Fifth repeat of item σ
F_1^σ
X_j^σ
X_5^σ
Working memory
F_0^σ
W_j^σ
W_5^σ
Node J = 5
wins competition
Summed input
to F_0^σ nodes
+
+
−
−
$[I_\sigma - \eta_+ j]^+$
$[\Lambda_\sigma - \eta_- j]^+$
I_σ
Λ_σ
Integrator
node
I_σ
I_σ
Θ_σ
+
−
+
σ
Item node σ
(b) Item σ repeated seven times
F_1^σ
X_j^σ
Working
memory
F_0^σ
W_j^σ
Activity
in WTA
F_0^σ inputs

Figure 11

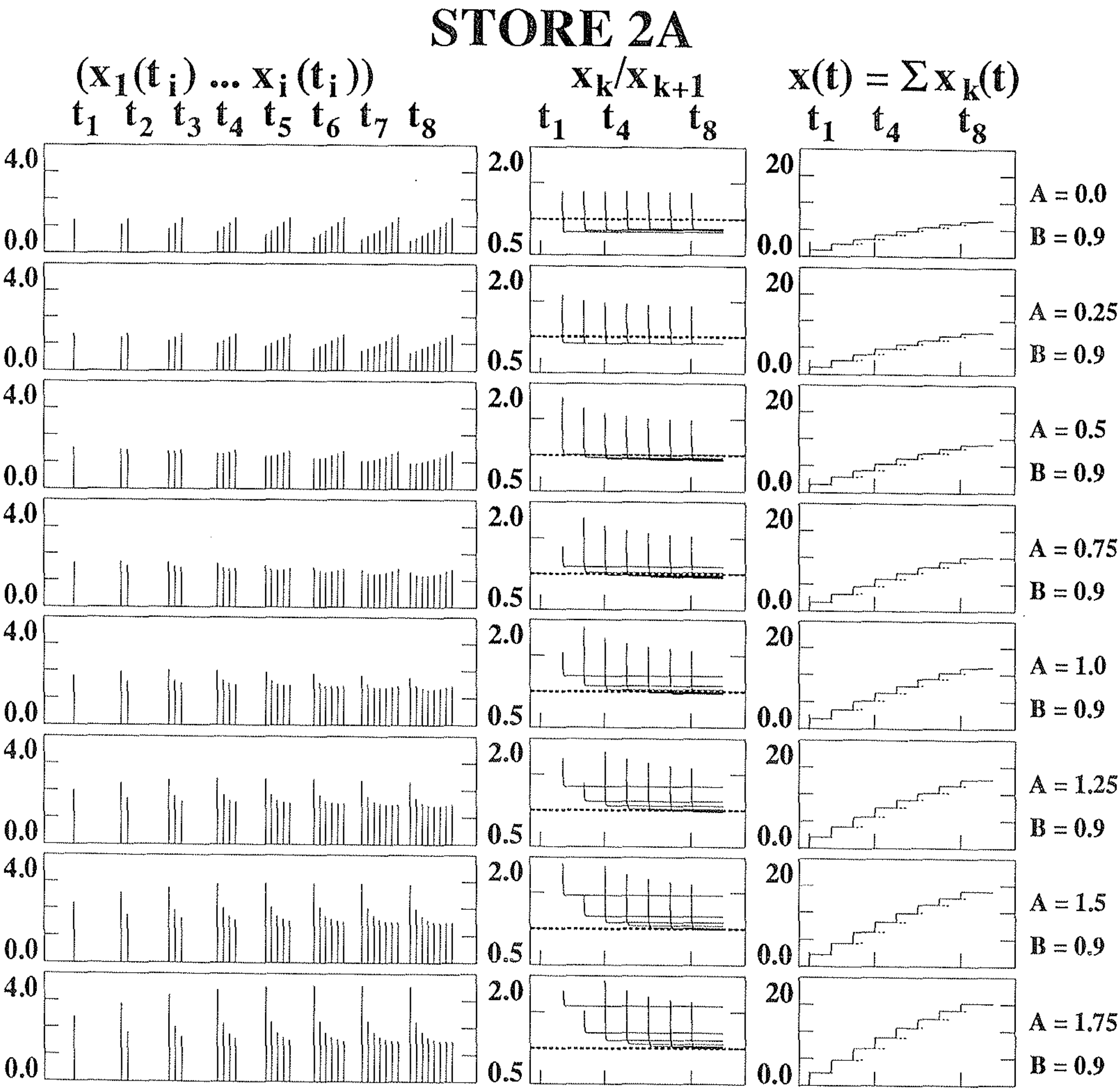

STORE 2A
(x1(ti) ... xi(ti))
xk/xk+1
x(t) = Σxk(t)
t1 t2 t3 t4 t5 t6 t7 t8
4.0
0.0
2.0
0.5
20
A = 0.0 B = 0.9
A = 0.25 B = 0.9
A = 0.5 B = 0.9
A = 0.75 B = 0.9
A = 1.0 B = 0.9
A = 1.25 B = 0.9
A = 1.5 B = 0.9
A = 1.75 B = 0.9

Figure 12

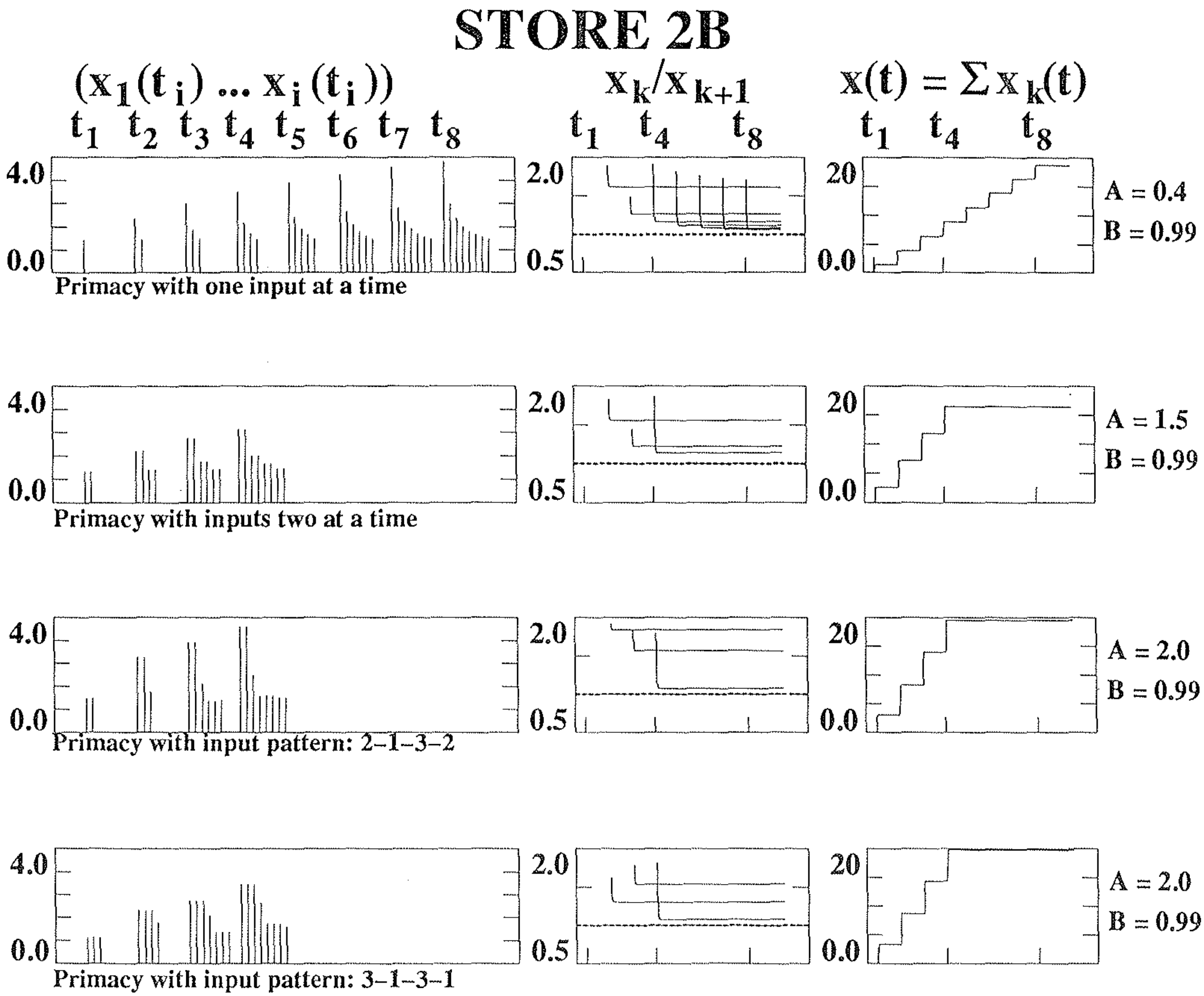

STORE 2B
$(x_1(t_i) \dots x_i(t_i))$
x_k/x_{k+1}
$x(t) = \Sigma x_k(t)$
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8
t_1 t_4 t_8
4.0
0.0
2.0
0.5
20
Primacy with one input at a time
A = 0.4
B = 0.99
Primacy with inputs two at a time
A = 1.5
B = 0.99
Primacy with input pattern: 2–1–3–2
A = 2.0
B = 0.99
Primacy with input pattern: 3–1–3–1
A = 2.0
B = 0.99

Figure 13

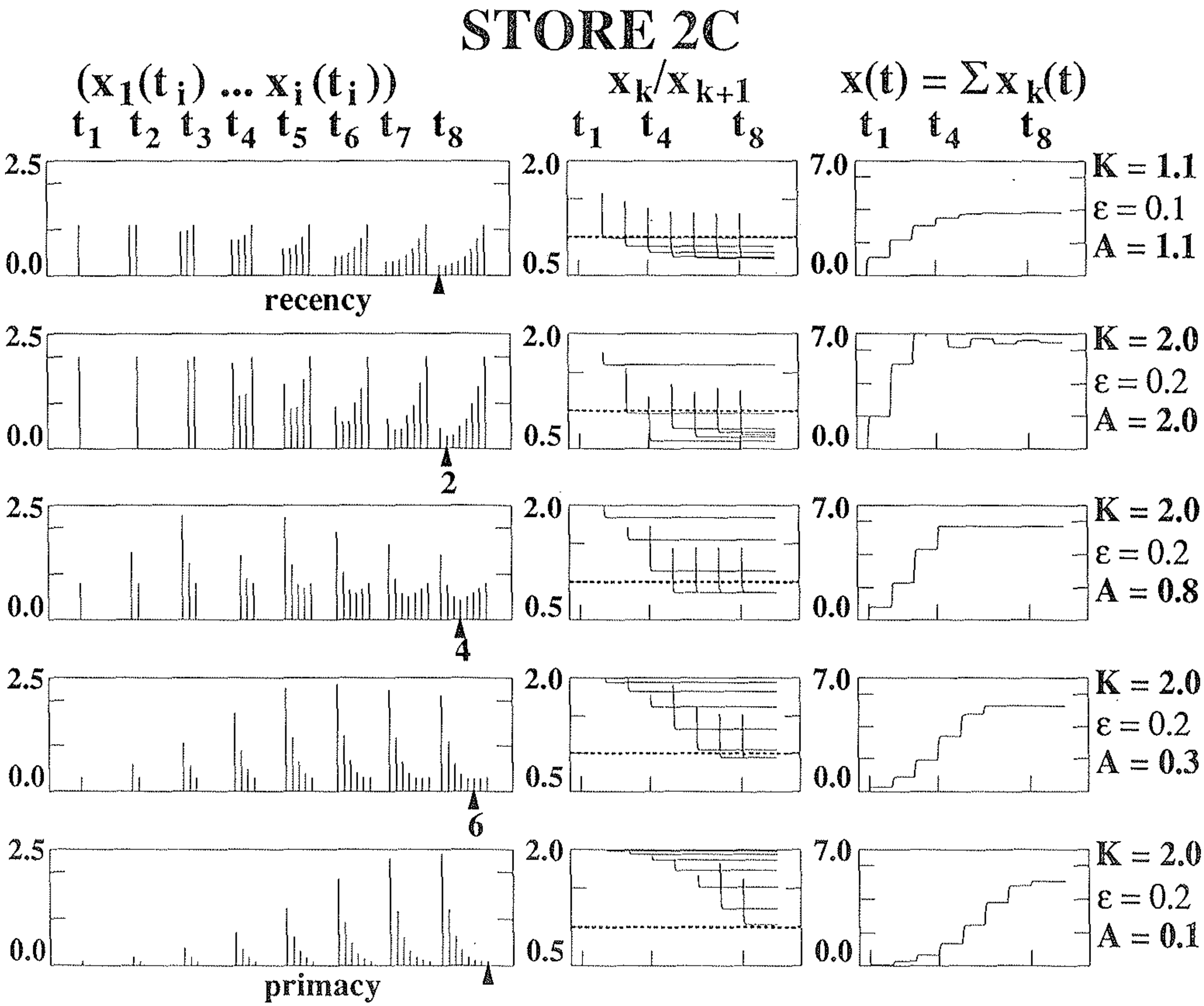

STORE 2C
(x1(ti) ... xi(ti))
xk/xk+1
x(t) = Σxk(t)
t1 t2 t3 t4 t5 t6 t7 t8
t1 t4 t8
2.5
0.0
2.0
0.5
7.0
K = 1.1
ε = 0.1
A = 1.1
recency
K = 2.0
ε = 0.2
A = 2.0
2
A = 0.8
4
A = 0.3
6
A = 0.1
primacy

Figure 14