Adaptive bidirectional associative memories

Bart Kosko

Bidirectionality, forward and backward information flow, is introduced in neural networks to produce two-way associative search for stored stimulus-response associations (A_i,B_i). Two fields of neurons, F_A and F_B, are connected by an $n \times p$ synaptic marix M. Passing information through M gives one direction, passing information through its transpose M^T gives the other. Every matrix is bidirectionally stable for bivalent and for continuous neurons. Paired data (A_i,B_i) are encoded in M by summing bipolar correlation matrices. The bidirectional associative memory (BAM) behaves as a two-layer hierarchy of symmetrically connected neurons. When the neurons in F_A and F_B are activated, the network quickly evolves to a stable state of two-pattern reverberation, or pseudoadaptive resonance, for every connection topology M. The stable reverberation corresponds to a system energy local minimum. An adaptive BAM allows M to rapidly learn associations without supervision. Stable short-term memory reverberations across F_A and F_B gradually seep pattern information into the long-term memory connections M, allowing input associations (A_i,B_i) to dig their own energy wells in the network state space. The BAM correlation encoding scheme is extended to a general Hebbian learning law. Then every BAM adaptively resonates in the sense that all nodes and edges quickly equilibrate in a system energy local minimum. A sampling adaptive BAM results when many more training samples are presented than there are neurons in F_A and F_B, but presented for brief pulses of learning, not allowing learning to fully or nearly converge. Learning tends to improve with sample size. Sampling adaptive BAMs can learn some simple continuous mappings and can rapidly abstract bivalent associations from several noisy gray-scale samples.

I. Introduction: Storing Data Pairs in Associative Memory Matrices

An $n \times p$ real matrix M can be interpreted as a matrix of synapses between two fields of neurons. The input or bottom-up field F_A consists of n neurons $\{a_1, \ldots, a_n\}$. The output or top-down field F_B consists of p neurons $\{b_1, \ldots, b_p\}$. The neurons a_i and b_j are the units of short-term memory (STM). For convenience, we shall use a_i and b_j to indicate neuron names and neuron states. Matrix entry m_{ij} is the synaptic connection from a_i to b_j. It is the unit of long-term memory (LTM). The sign of m_{ij} determines the type of synaptic connection: excitatory if $m_{ij} > 0$, inhibitory if $m_{ij} < 0$. The magnitude of m_{ij} determines the strength of the connection. A real n-dimensional row vector $\mathbf{A}$ represents a state of F_A, a STM pattern of activity across the neurons $a_1, \ldots, a_n$. A real p-dimensional row vector $\mathbf{B}$ represents a state of F_B. An associative memory is any vector space transformation $T:R^n \rightarrow R^p$. Usually T is nonlinear. The matrix mapping $M:R^n \rightarrow R^p$ is a linear associative memory. When F_A and F_B are distinct, M is a heteroassociative associative memory. It stores vector data pairs $(\mathbf{A}_i,\mathbf{B}_i)$. In the special case when $F_A = F_B$, M is an autoassociative associative memory. It stores data vectors $\mathbf{A}_i$.

Recall proceeds through vector–matrix multiplication and nonlinear state transition. The p-vector $\mathbf{A}\,\mathbf{M}$ is a fan-in vector of input sums to the neurons in F_B: $\mathbf{A}\,\mathbf{M} = (I_{b1}, \ldots, I_{bp})$. Specifically, each neuron a_i fans out its numeric output a_i across each synaptic pathway m_{ij}, sending the gated product $a_i\, m_{ij}$ to each neuron b_j in F_B. Each neuron b_j receives a fan-in of n gated products $a_i\, m_{ij}$, arriving independently and perhaps asynchronously, and sums them to compute its input $I_{bj} = a_1\, m_{1j} + \ldots + a_n\, m_{nj}$. Neuron b_j processes input I_{bj} to produce the output signal $S(I_{bj})$. In general the signal function S is nonlinear, usually sigmoidal or S-shaped. The associative memory M recalls the vector of output signals $[S(I_{b1}), \ldots, S(I_{bp})]$ when presented with input key A. In the simplest associative memories, linear associative memories, each neuron's output signal is simply its input signal: $S(I_{bj}) = I_{bj}$. Then associative recall is simply vector multiplication: $\mathbf{B} = \mathbf{A}\,\mathbf{M}$.

What is the simplest way to store m data pairs $(A_1,B_1),(A_2,B_2), \ldots, (A_m,B_m)$ in an $n \times p$ associative memory matrix M? The simplest storage procedure is to convert each association (A_i,B_i) into an $n \times p$ matrix M_i, then combine each association matrix M_i pointwise. The simplest pointwise combination technique is addition: $M = M_1 + \ldots + M_m$. The simplest operation for converting two row vectors $\mathbf{A}_i$ and $\mathbf{B}_i$ of dimensions n and p into an $n \times p$ matrix M_i is the vector outer product $\mathbf{A}_i^T\mathbf{B}_i$. So the simplest way to store m $(\mathbf{A}_i,\mathbf{B}_i)$ is to sum outer product or correlation matrices:

$$M = \mathbf{A}_1^T\mathbf{B}_1 + \ldots + \mathbf{A}_m^T\mathbf{B}_m. \tag{1}$$

This is the familiar storage method used in the theory of linear associative memories, studied by Kohonen[1,2] and Anderson *et al.*[3] If the input patterns $\mathbf{A}_1, \ldots, \mathbf{A}_m$

The author is with University of Southern California, Department of Electrical Engineering—Systems, Los Angeles, California 90089.

Received 27 May 1987.
0003-6935/87/234947-14$02.00/0.

are orthonormal—$\mathbf{A}_i\mathbf{A}_j^T = 1$ if $i = j$, 0 if not—perfect recall of the associated output patterns $\{\mathbf{B}_1, \ldots, \mathbf{B}_m\}$ is achieved in the forward direction:

$$\mathbf{A}_i\mathrm{M} = \mathbf{A}_i\mathbf{A}_i^T\mathbf{B}_i + \sum_{i \neq j} (\mathbf{A}_i\mathbf{A}_j^T)\mathbf{B}_j$$

$$= \mathbf{B}_i. \tag{2}$$

If $\mathbf{A}_1, \ldots, \mathbf{A}_m$ are not orthonormal, as in general they are not, the second term on the right-hand side of Eq. (2), the noise term, contributes crosstalk to the recalled pattern by additively modulating the signal term. More generally, as Kohonen[2] has shown, the least-squares optimal linear associative memory (OLAM) M is given by $\mathrm{M} = \mathrm{A}^* \mathrm{B}$, where A is the $m \times n$ matrix whose ith row is A_i, B is the $m \times p$ matrix whose ith row is B_i, and A^* is the Moore-Penrose pseudoinverse of A. If $\{\mathrm{A}_1, \ldots, \mathrm{A}_m\}$ are orthonormal, the OLAM $\mathrm{M} = \mathrm{A}^T\mathrm{B}$, which is equivalent to the memory scheme in Eq. (1).

II. Discrete Bidirectional Associative Memory (BAM) Stability

Suppose we wish to synchronously feed back the recalled output B to an associative memory M to improve recall accuracy. The recalled output B is some nonlinear transformation S of the input sum $\mathrm{A\,M{:}B} = S(\mathrm{A\,M}) = [S(\mathrm{AM}^1), \ldots, S(\mathrm{A\,M}^p)]$, where M^j is the jth column of M. What is the simplest way to feed B back to the associative memory? Since M has dimensions $n \times p$ and $\mathbf{B}$ is a p vector, $\mathbf{B}$ cannot vector multiply M, but it can multiply the M matrix transpose (adjoint) M^T. Thus the simplest feedback scheme is to pass $\mathbf{B}$ backward through M^T. Any other feedback scheme requires more information in the form of a $p \times n$ matrix N different from M^T. Field F_A receives the top-down message $\mathbf{B}\,\mathrm{M}^T$ and produces the new STM pattern $A' = S(\mathbf{B}\,\mathrm{M}^T) = [S(\mathbf{B}\,\mathrm{M}_1^T), \ldots, S(\mathbf{B}\,\mathrm{M}_n^T)]$ across F_A, where M_i is the ith row (column) of M (M^T). Carpenter[4] and Grossberg[5–9] interpret top-down signals as expectations in their adaptive resonance theory (ART). Intuitively A' is what the field F_B expects to see when it receives bottom-up input B.

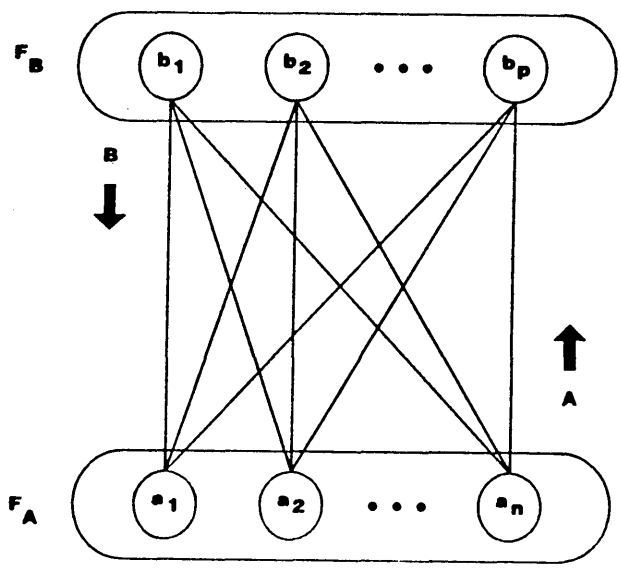
If A' is fed back through M, a new B' results, which can be fed back through M^T to produce A'', and so on. Ideally this back-and-forth flow of distributed information will quickly equilibrate or resonate on a fixed data pair (A_f, B_f):

$$\begin{array}{lllll} A & \rightarrow & \mathrm{M} & \rightarrow & B, \\ A' & \leftarrow & \mathrm{M}^T & \leftarrow & B, \\ A' & \rightarrow & \mathrm{M} & \rightarrow & B', \\ A'' & \leftarrow & \mathrm{M}^T & \leftarrow & B', \\ & & \vdots & & \\ A_f & \rightarrow & \mathrm{M} & \rightarrow & B_f, \\ A_f & \leftarrow & \mathrm{M}^T & \leftarrow & B_f, \\ & & \vdots & & \end{array}$$

If an associative memory matrix M equilibrates in this fashion for every input pair (A,B), then M is said to be bidirectionally stable.[10,11]

Which matrices are bidirectionally stable for which signal functions S? Linear associative memory matrices are obviously in general not bidirectionally stable. We shall limit our discussion to sigmoidal or S-shaped signal functions S, such as $S(x) = (1 + e^{-x})^{-1}$, or more generally, to bounded monotone increasing signal functions. Grossberg[12] long ago showed that this is not a limitation at all. He proved that, roughly speaking, a sigmoidal signal function is optimal in the sense that, in unidirectional competitive networks, it computes a quenching threshold below which neural activity is suppressed as noise and above which activity is contrast enhanced and then stored as a stable reverberation in STM. In particular, linear signal functions amplify noise as faithfully as they amplify signals. This theoretical fact reflects the evolutionary fact that real neuron firing frequency is sigmoidal.

First we consider bivalent, or McCulloch-Pitts,[13] neurons. Each neuron a_i and b_j is either on (+1) or off (0 or −1) at any time. Hence a state A of F_A is a point in the Boolean n-cube $B^n = \{0,1\}^n$ or $\{-1,1\}^n$. A state B of F_B is a point in $B^p = \{0,1\}^p$ or $\{-1,1\}^p$. A state of the bidirectional associative memory (BAM) (F_A, M, F_B) is a point (A,B) in the bivalent product space $B^n \times B^p$. Topologically, a BAM can be viewed as a two-layer hierarchy of symmetrically connected fields:



What is the simplest signal function S for a bivalent BAM (F_A, M, F_B)? The simplest S is a threshold function:

$$a_i = \begin{cases} 1 & \text{if } \mathbf{B}\,M_i^T > 0, \\ 0 & \text{if } \mathbf{B}\,M_i^T < 0, \end{cases} \tag{3}$$

$$b_j = \begin{cases} 1 & \text{if } \mathbf{A}\,\mathrm{M}^j > 0, \\ 0 & \text{if } \mathbf{A}\,\mathrm{M}^j < 0, \end{cases} \tag{4}$$

where once again M_i is the ith row (column) of M (M^T)

and M^j is the jth column (row) of M (M^T). If the input sum to each neuron equals its threshold 0, the neuron maintains its current state. It stays on if it already is on, off if already off. For simplicity, each neuron has threshold 0 and no external inputs. In general, a_i has a numeric threshold T_i and constant numeric input I_i; b_j has threshold S_j and input J_j. A bivalent BAM is then specified by the vector 7-tuple (F_A, T, I, M, F_B, S, J) and the threshold laws (3) and (4) are modified accordingly; e.g., $a_i = 1$ if $\mathbf{B}\,M_i^T + I_i > T_i$.

Which matrices M are bidirectionally stable for bivalent BAMs? All matrices. Every synaptic connection topology rapidly equilibrates, no matter how large the dimensions n and p. This surprising theorem is proved in Refs. 11 and 14 and generalizes the well-known undirectional stability for autoassociative networks with square symmetric M, as popularized by Hopfield[15] and reviewed below. Bidirectionality, forward and backward information flow, in neural nets produces two-way associative search for the nearest stored pair ($\mathbf{A}_i$,$\mathbf{B}_i$) to an input key. Since every matrix is bidirectionally stable, many more matrices can be decoded than those in which information has been deliberately encoded.

When the BAM neurons are activated, the network quickly evolves to a stable state of two-pattern reverberation, or nonadaptive resonance.[4,7] The resonance is nonadaptive because no learning occurs. The weights m_{ij} are fixed. This behavior approximates equilibrium behavior in a learning context since changes in the synapses (LTM traces) m_{ij} are invariably slower than changes in the neuron activations (STM traces) a_i and b_j. Below we shall exploit this property to construct adaptive BAMs.

The stable reverberation corresponds to a system energy local minimum. Geometrically, an input pattern is placed on the BAM energy surface as a ball bearing in the bivalent product space $B^n \times B^p$. In particular, the bipolar correlation encoding scheme described below sculpts the energy surface so that the data pairs (A_i,B_i) are stored as local energy minima. The input ball bearing rolls down into the nearest basin of attraction, dissipating energy as it rolls. Frictional damping brings it to rest at the bottom of the energy well, and the pattern is classified or misclassified accordingly. Thus the BAM behaves as a programmable dissipative dynamic system.

For completeness we review the proof[10,11] that every matrix is bivalently bidirectionally stable. The proof technique is to show that some system functional E:$B^n \times B^p \rightarrow R$ is a Lyapunov function or bounded monotone decreasing energy function for the network. The energy function decreases if state changes occur. System stability occurs when the functional E rapidly obtains its lower bound, where it stays forever. Lyapunov functionals provide a shortcut to the global analysis of nonlinear dynamic systems, sidestepping the often hopeless task of solving the many coupled nonlinear difference or differential equations. The most general Lyapunov stability result is the Cohen-Grossberg theorem[16] for symmetric undirectional autoassociators, which we extend in this and the next section to arbitrary bidirectional heteroassociators. The Lyapunov trick of the Cohen-Grossberg theorem is to substitute the neuron state-transition equations into the derivative of the appropriate energy function, and then use a sign argument to show that the derivative is always nonpositive. Hopfield[15] used the discrete version of this Lyapunov trick to show that zero-diagonal symmetric unidirectional autoassociators are stable for asynchronous or serial state changes, i.e., where at any moment at most one neuron changes state. The argument we now present subsumes this case when $F_A = F_B$ and $M = M^T$ in simple asynchronous operation. An appropriate measure of the energy of the bivalent (A,B) is the sum (average) of two energies: the energy $A\,M\,B^T$ of the forward pass and the energy $B\,M^T\,A^T$ of the backward pass. Taking the negative of these quadratic forms gives

$$\begin{aligned} E(\mathrm{A,B}) &= -\tfrac{1}{2}\,\mathrm{A\,M\,B}^T - \tfrac{1}{2}\,\mathrm{B\,M}^T\mathrm{A}^T \\ &= -\,\mathrm{A\,M\,B}^T \\ &= -\sum_i \sum_j a_i b_j m_{ij}, \end{aligned} \tag{5}$$

provided all thresholds $T_i = S_j = 0$ and inputs $I_i = J_j = 0$, which we shall assume for simplicity. In general the appropriate energy function includes thresholds and inputs linearly:

$$E(\mathrm{A,B}) = \mathrm{A\,M\,B}^T - I\,\mathrm{A}^T + T\,\mathrm{A}^T - J\,\mathrm{B}^T + S\,\mathrm{B}^T.$$

BAM convergence is proved by showing that synchronous or asynchronous state changes decrease the energy and that the energy is bounded below, so the BAM monotonically gravitates to fixed points. E is trivially bounded below for all A and B:

$$E(\mathrm{A,B}) \geq -\sum_i \sum_j |m_{ij}|.$$

Synchronous vs asynchronous state changes must be clarified. Synchronous behavior occurs when all or some neurons within a field change their state at the same clock cycle. Asynchronous behavior is a special case. Simple asynchronous behavior occurs when only one neuron per field changes state per cycle. Subset asynchronous behavior occurs when some proper subset of neurons within a field changes state per cycle. These definitions of asynchrony are cross sectional. The resultant time-series interpretation of asynchronous behavior is that each neuron in a field randomly and independently changes state, converting the BAM network into a stochastic process. In the proof below we do not assume that changes occur concurrently in the two fields F_A and F_B. Otherwise, in principle the energy function might increase. Examination of the argument below shows, though, that this is very unlikely in large networks since so many additive terms in the energy differential are always negative. In any event, the BAM model of back-and-forth information flow we have been developing implicitly assumes that state changes are occurring in at most one field F_A or F_B at a

time. Further, the Lyapunov argument below shows that synchronous operation produces sums of pointwise (neuronwise) energy changes that can be large. In practice this means synchronous updates produce much faster convergence than asynchronous updates.

First we consider state changes in field F_A. A similar argument will hold for changes in F_B. Field F_A change is denoted by $\Delta A = A_2 - A_1 = (\Delta a_1, \ldots, \Delta a_n)$ and energy change by $\Delta E = E_2 - E_1$. Hence $\Delta a_i = -1$, 0, or +1 for a binary neuron. Then

$$\begin{aligned}\Delta E &= -\Delta \mathrm{A}\, \mathrm{M}\, \mathrm{B}^T \\ &= -\sum_i \Delta a_i \sum_j b_j m_{ij} \\ &= -\sum_i \Delta a_i B\, M_i^T. \end{aligned} \tag{6}$$

We need only consider nonzero state changes. If $\Delta a_i > 0$, the state transition law (3) above implies $\mathrm{B}\, \mathrm{M}_i^T > 0$. If $\Delta a_i < 0$, Eq. (3) implies $\mathrm{B}\, \mathrm{M}_i^T < 0$. Hence state change and input sum agree in sign. Hence their product is positive: $\Delta a_i\, \mathrm{B}\, \mathrm{M}_i^T > 0$. Hence $\Delta E < 0$. Similarly, the sign law (4) for b_j implies $\Delta E = -A\, \mathrm{M}\, \Delta \mathrm{B}^T < 0$. Since M was an arbitrary $n \times p$ real matrix, this proves that every matrix is bivalently bidirectionally stable.

III. BAM Correlation Encoding

Which BAM matrix M best encodes m binary pairs (A_i,B_i)? The correlation encoding scheme in Eq. (1) suggests adding the outer-product matrices $\mathrm{A}_i^T\, \mathrm{B}_i$ pointwise, at least to facilitate forward recall. Will this work for backward recall? The linearity of the transpose operator implies that it will:

$$\begin{aligned}\mathrm{M}^T &= (\mathrm{A}_1^T \mathrm{B}_1)^T + \ldots (\mathrm{A}_m^T \mathrm{B}_m)^T \\ &= \mathrm{B}_1^T \mathrm{A}_1 + \ldots + \mathrm{B}_m^T \mathrm{A}_m. \end{aligned} \tag{7}$$

However, the additive scheme (1) implies that if we use only binary vectors, M will contain no inhibitory synapses. So the input sums $\mathrm{B}\, \mathrm{M}_i^T$ and $\mathrm{A}\, \mathrm{M}^j$ can never be negative. So the state transition laws (3) and (4) imply that $a_i = b_j = 1$ once a_i and b_j turn on, which they probably will after the first update. Exceptions can occur for initial null vectors or a null matrix M, when $a_i = b_j = 0$.

Bipolar state vectors do not produce this problem. Suppose (X_i,Y_i) is the bipolar version of the binary pair (A_i,B_i), i.e., binary zeros are replaced with minus ones, i.e., $X_i = 2A_i - I$ and $Y_i = 2B_i - I$, where I is a unit vector of n-many or p-many ones. Then the ijth entry of $\mathrm{X}_k^T \mathrm{Y}_k$ is excitatory (+1) if the vector elements x_i^k and y_j^k agree in sign, inhibitory (−1) if they disagree in sign. This is simple conjunctive or Hebbian correlation learning. Thus the sum M of bipolar outer-product matrices

$$\mathrm{M} = \mathrm{X}_1^T \mathrm{Y}_1 + \ldots + \mathrm{X}_m^T \mathrm{Y}_m \tag{8}$$

naturally weights the excitatory and inhibitory connections. Multiplying M or M^T by binary or bipolar vectors produces input sums of different signs, so Eqs. (3) and (4) are not trivialized.

Note that to encode m binary vectors $\mathbf{A}_1, \ldots, \mathbf{A}_m$ in a unidirectional autoassociative memory matrix, Eq. (8) reduces to the symmetric matrix $\mathrm{X}_1^T \mathrm{X}_1 + \ldots + \mathrm{X}_m^T \mathrm{X}_m$, which is the storage mechanism used by Hopfield[15] (who also zeros the main diagonal to improve recall). Note also that the pair $(\mathrm{A}_i,\mathrm{B}_i)$ can be unlearned or forgotten (erased) by summing $-\mathrm{X}_i^T \mathrm{Y}_i$, or, equivalently, by encoding $(\mathrm{A}_i^c,\mathrm{B}_i)$ or $(\mathrm{A}_i,\mathrm{B}_i^c)$ since bipolar complements are given by $\mathrm{X}_i^c = -\mathrm{X}_i$ and $\mathrm{Y}_i^c = -\mathrm{Y}_i$. Equation (8) allows data to be read, written, or erased from memory. Further, $(\mathrm{X}_i^c)^T\, \mathrm{Y}_i^c = \mathrm{X}_i^T \mathrm{Y}_i$, so storing $(\mathrm{A}_i,\mathrm{B}_i)$ through Eq. (8) implies storing $(\mathrm{A}_i^c,\mathrm{B}_i^c)$ as well.

Strictly speaking bipolar correlation learning laws such as Eq. (8) can be biologically implausible. They imply that synapses can change character from excitatory to inhibitory, or inhibitory to excitatory, with successive experience. This is seldom observed with real synapses. However, when the number of stored patterns m is fairly large, $|m_{ij}| > 0$ tends to hold. So the addition or deletion of relatively few patterns does not on average change the sign of m_{ij}.

Is it better to use binary or bipolar state vectors for recall from Eq. (8)? In Ref. 10 we prove that bipolar coding is better on average. Much of the argument can be seen from the properties of the bipolar signal–noise expansion

$$\begin{aligned}\mathrm{X}_i\mathrm{M} &= (\mathrm{X}_i\mathrm{X}_i^T)\mathrm{Y}_i + \sum_{j \neq i} (\mathrm{X}_i\mathrm{X}_j^T)\mathrm{Y}_j \\ &= n\mathrm{Y}_i + \sum_{j \neq i} (\mathrm{X}_i\mathrm{X}_j^T)\mathrm{Y}_j \\ &= \sum_j c_{ij}\mathrm{Y}_j, \end{aligned} \tag{9}$$

where $c_{ij} = c_{ji} = \mathrm{X}_i\, \mathrm{X}_j^T$.

The c_{ij} are correction coefficients. Ideally the c_{ij} will behave in sign and magnitude so as to move Y_j closer to Y_i and give Y_j more positive weight the closer Y_j is to Y_i. Then the right-hand side of Eq. (9) will tend to equal a positive multiple of Y_i and thus threshold to Y_i or B_i. When the input X is nearer X_i than all other X_j, the subsequent output Y should tend to be nearer Y_i than all other Y_j. When Y is fed back through M^T, the output X′ should tend to be even closer to X_i than X was, and so on. Combining this argument with the signal–noise expansion (9) and its transpose-based backward analog, we obtain an estimate of the BAM storage capacity for reliable recall: $m < \min(n,p)$. No more data pairs can be stored and accurately recalled than the lesser of the vector dimensions used.

This analysis explains much BAM behavior without Lyapunov techniques. However, such accurate decoding implicitly assumes that if stored input patterns are close, stored output patterns are close. Specifically we make the continuity assumption:

$$1/nH(A_i,A_j) \sim 1/pH(B_i,B_j), \tag{10}$$

where H(. . .) denotes Hamming or l^1 distance. This is an implicit assumption of continuous mapping networks. When a data set substantially violates it, as in the parity mapping, which indicates whether there is an even or odd number of ones in a bit vector, supervised learning techniques such as backward error propagation[17–20] are preferable.

Do the correction coefficients c_{ij} behave as desired? They do, when (10) holds, in the sense that they naturally connect bipolar and binary spaces:

$$c_{ij} \lesseqgtr 0 \quad \text{iff} \quad H(A_i,A_j) \gtreqless n/2. \tag{11}$$

Expression (11) follows from

$$\begin{aligned} c_{ij} &= X_i X_j^T \\ &= (\text{number of common elements}) \\ &\quad - (\text{number of different elements}) \\ &= [n - H(A_i,A_j)] - H(A_i,A_j) \\ &= n - 2H(A_i,A_j). \end{aligned} \tag{12}$$

If A_j is more than half the space away, so to speak, from A_i, and thus by (10) if B_j is approximately more than half the space away from B_i, the negative sign of c_{ij} corrects Y_j by converting it to Y_j^c, which is a better approximation of Y_i since B_j^c is approximately less than half the space away from B_i. The magnitude of c_{ij} then further corrects Y_j by directly approaching the maximum signal amplification factor, n, as $H(B_iB_j^c)$ approaches 0. If A_j is less than half the space away from A_i, then $c_{ij} > 0$ and c_{ij} approaches n as $H(B_i,B_j)$ approaches 0. If A_j is equidistant between A_i and A_i^c, then $c_{ij} = 0$. Finally, bipolar coding of state vectors is better on average than binary coding in the sense that on average

$$A_i X_i^T \gtreqless c_{ij} \quad \text{iff} \quad H(A_i,A_j) \gtreqless n/2 \tag{13}$$

tends to hold. So on average the c_{ij} always correct better in magnitude than the mixed coefficients $A_i X_j^T$ and sometimes the mixed coefficients can have the wrong sign.

Consider a simple example. Suppose we wish to store two pairs given by

$$A_1 = (1\,0\,1\,0\,1\,0) \qquad B_1 = (1\,1\,0\,0),$$

$$A_2 = (1\,1\,1\,0\,0\,0) \qquad B_2 = (1\,0\,1\,0).$$

Note that the vectors are nonorthogonal and that the continuity assumption (10) holds since $1/6\ H(A_1,A_2) = 1/3 \sim 1/2 = 1/4\ H(B_1,B_2)$. Convert these binary pairs to bipolar pairs:

$$X_1 = (1\ \ -1\ \ 1\ \ -1\ \ 1\ \ -1) \qquad Y_1 = (1\ \ 1\ \ -1\ \ -1),$$

$$X_2 = (1\ \ 1\ \ 1\ \ -1\ \ -1\ \ -1) \qquad Y_2 = (1\ \ -1\ \ 1\ \ -1).$$

Convert the bipolar pairs (X_i,Y_i) to correlation matrices $X_i^T Y_i$:

$$X_1^T Y_1 = \begin{pmatrix} 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \end{pmatrix},$$

$$X_2^T Y_2 = \begin{pmatrix} 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 \end{pmatrix}.$$

Then M is given by $M = X_1^T Y_1 + X_2^T Y_2$:

$$M = \begin{pmatrix} 2 & 0 & 0 & -2 \\ 0 & -2 & 2 & 0 \\ 2 & 0 & 0 & -2 \\ -2 & 0 & 0 & 2 \\ 0 & 2 & -2 & 0 \\ -2 & 0 & 0 & 2 \end{pmatrix}.$$

Then, using binary vectors for recall for ease of computing, we see that

$$A_1M = (4\ \ 2\ \ -2\ \ -4) \rightarrow (1\ \ 1\ \ 0\ \ 0) = B_1,$$

$$A_2M = (4\ \ -2\ \ 2\ \ -4) \rightarrow (1\ \ 0\ \ 1\ \ 0) = B_2,$$

on using the threshold signal function (4) and, on using Eq. (3),

$$B_1M^T = (2\ \ -2\ \ 2\ \ -2\ \ 2\ \ -2) \rightarrow (1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0) = A_1,$$

$$B_2M^T = (2\ \ 2\ \ 2\ \ -2\ \ -2\ \ -2) \rightarrow (1\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0) = A_2.$$

The use of synchronous updates, combined with satisfying the continuity assumption and the memory capacity constraint [2 < min(6,4)], produced instant convergence to the local energy minima $E(A_1,B_1) = -A_1 M B_1^T = -(4\,2\,-2\,-4)(1\,1\,0\,0)^T = -6 = E(A_2,B_2)$. Suppose we perturb A_2 by 1 bit. In particular, suppose we present an input A = (0 1 1 0 0 0) to the BAM. Then

$$A\,M = (2\ \ -2\ \ 2\ \ -2) \rightarrow (1\ \ 0\ \ 1\ \ 0) = B_2,$$

and thus A evokes the resonant pair (A_2,B_2) with initial energy $E(A,B_2) = -4$. Now suppose an input A = (0 0 0 1 1 0) is presented to the BAM. Since $H(A,A_1) = 3 < 5 = H(A,A_2)$, we might expect A to evoke the resonant pair (A_1,B_1). In fact

$$A\,M = (-2\ \ 2\ \ -2\ \ 2) \rightarrow (0\ \ 1\ \ 0\ \ 1) = B_2^c,$$

and B_2^c in turn recalls A_2^c, which recalls B_2^c, etc., with energies $E(A,B_2^c) = -4 > -6 = E(A_2^c,B_2^c)$ since $H(A,A_2^c) = 1$. We recall that the bipolar correlation encoding scheme (8) stores (A_i^c,B_i^c) when it stores (A_i,B_i).

Figure 1 displays snapshots of asynchronous BAM recall. Approximately six neurons update between snapshots. The spatial alphabetic associations (S,E), (M,V), and (G,N) are stored. F_A contains $n = 10 \times 14 = 140$ neurons. F_B contains $p = 9 \times 12 = 108$ neurons. A 40% noise corrupted version (99 bits randomly flipped) of (S,E) is presented to the BAM and (S,E) is perfectly recalled, illustrating the global order-from-

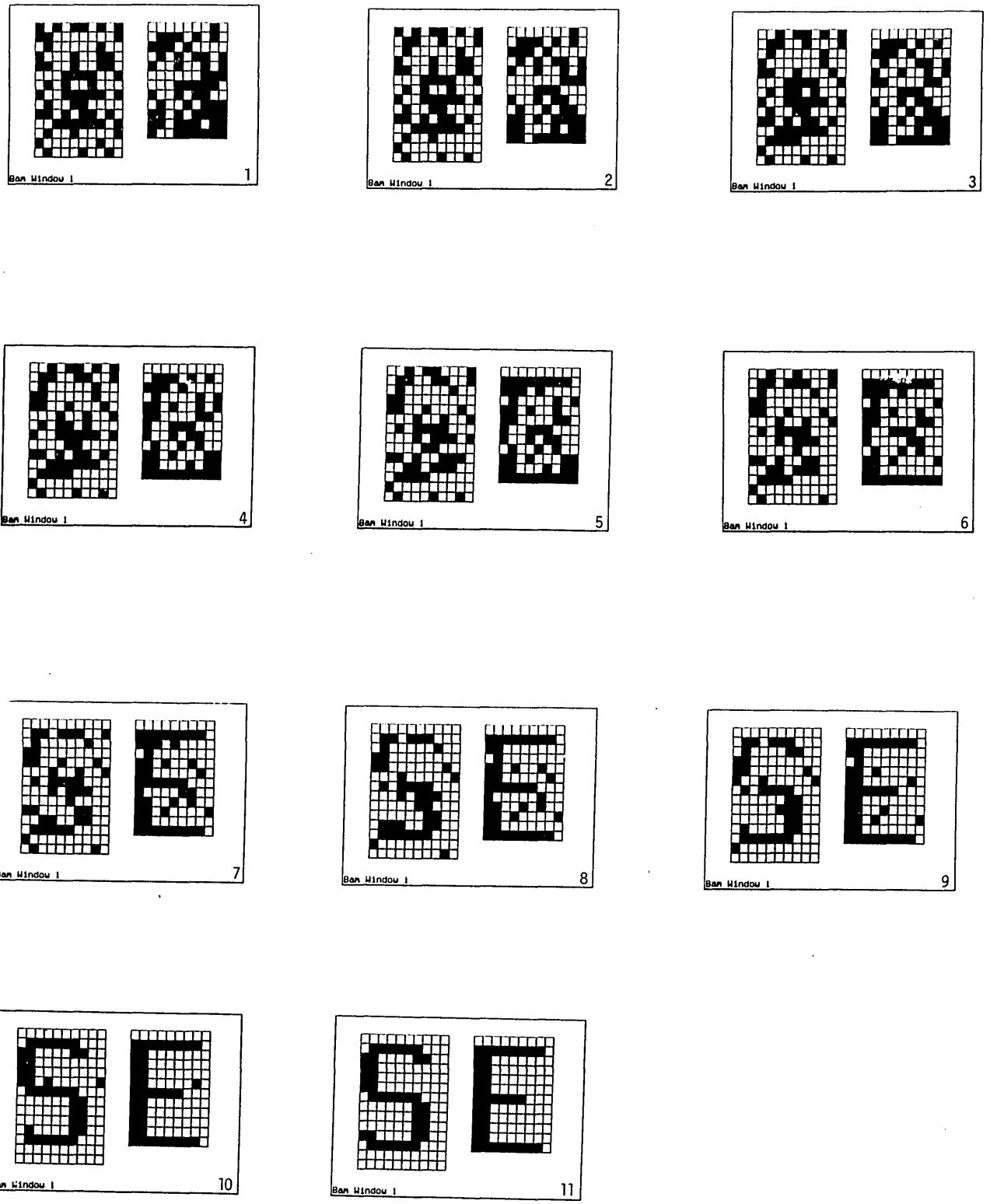


Fig. 1. Asynchronous BAM recall. Approximately six neurons update per snapshot. The associated spatial patterns (S,E), (M,V), and (G,N) are stored. Field F_A contains 140 neurons; F_B,108. Perfect recall of (S,E) is achieved when recall is initiated with a 40% noise-corrupted version of (S,E).

chaos aesthetic appeal of asynchronous BAM operation.

BAMs are also natural structures for optical implementation. Perhaps the simplest all-optical implementation is a holographic resonator with M housed in a transmission hologram sandwiched between two phase-conjugate mirrors. Figures 2 and 3 display two different optical BAMs discussed in Ref. 21. Figure 2 displays a simple matrix–vector multiplier BAM with M represented by a 2-D grid of pixels with varying transmittances. Figure 3 displays a BAM based on a volume reflection hologram. The box labeled threshold device accepts a weak signal image on one side and produces an intensified and contrast-enhanced version of the image on its output side. The Hughes liquid crystal light valve or two-wave mixing are two ways to implement such a device. Note that the configuration requires the hologram to be read with light of two different polarizations. Hence diffraction efficiency of holograms recorded as birefringence patterns in photorefractive crystals will be somewhat compromised.

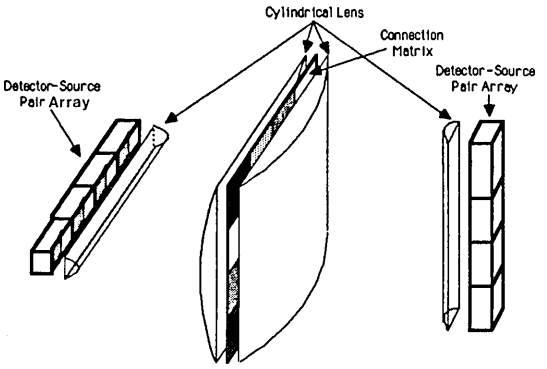


Fig. 2. Matrix–vector multiplier BAM.

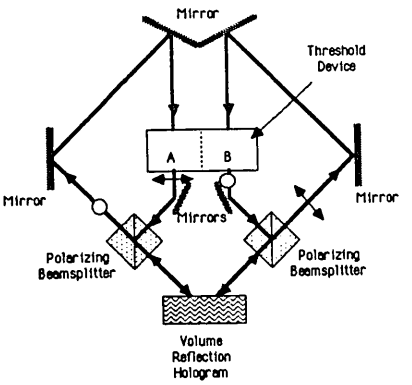


Fig. 3. BAM volume reflection hologram.

IV. Continuous BAMs

A continuous BAM[10,11] is specified by, for example, the additive dynamic system

$$\dot{a}_i = -a_i + \sum_j S(b_j)m_{ij} + I_i, \tag{14}$$

$$\dot{b}_j = -b_j + \sum_i S(a_i)m_{ij} + J_j, \tag{15}$$

where the overdot denotes time differentiation. The activations a_i and b_j can take on arbitrary real values. S is a sigmoid signal function. More generally, we shall only assume that S is bounded and strictly monotone increasing, so that $S' = dS(x)/dx > 0$. For definiteness, we assume all signals $S(x)$ are in [0,1] or [−1,1], so that the output (observable) state of the BAM is a trajectory in the product unit hypercube $I^n \times I^p$ where $I^n = [0,1]^n$ or $[-1,1]^n$. For example, in the simulations below we use the bipolar logistic sigmoid $S(x) = 2(1 + e^{-cx})^{-1} - 1$ for $c > 0$. I_i and J_j are constant external inputs.

The first term on the right-hand sides of Eqs. (14) and (15) are STM passive decay terms. The second term is the endogenous feedback term. It sums gated bipolar signals from all neurons in the opposite field. The third term is the exogenous input, which is assumed to change so slow relative to the STM reaction times that it is constant. Of course both right-hand sides of Eqs. (14) and (15) are in general multiplied by time constants, as is each term. We omit these constants for notational convenience.

The additive model [Eqs. (14) and (15)] can be extended to a shunting[8] or multiplicative model that allows multiplicative self-excitation through the term $(A_i - a_i)\,[S(a_i) + I_i^E]$ and multiplicative cross-inhibition through a similar term, where A_i (B_j) is the positive upper bound on the activation of a_i (b_j), and I_i^I (J_j^I) and I_i^E (J_j^E) are the respective constant non-negative inhibitory and excitatory inputs to a_i (b_j). The shunting model can then be written

$$\dot{a}_i = -a_i + (A_i - a_i)[S(a_i) + I_i^E] - a_i\left[\sum_j m_{ij}S(b_j) + I_i^I\right], \tag{16}$$

$$\dot{b}_j = -b_j + (B_j - b_j)[S(b_j) + J_j^E] - b_j\left[\sum_i m_{ij}S(a_i) + J_j^I\right]. \tag{17}$$

The inhibitory shunt a_i (b_j) can be replaced with $C_i + a_i$ ($D_j + b_j$) where C_i (D_j) is a non-negative constant. Then the range of a_i (b_j) is the interval $[-C_i, A_i]$ ($[-D_j, B_j]$). The bidirectional stability of systems (16) and (17) follows from the same source of stability as the additive model, the bidirectional/heteroassociative extension of the Cohen-Grossberg theorem.[16] The thrust of this extension is to symmetrize an arbitrary rectangular connection matrix M by forming the zero-block diagonal matrix N:

$$\begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix},$$

so that $N = N^T$. Thus the bidirectional heteroassociative procedure is converted to a large-scale unidirectional autoassociative procedure acting on the augmented state vectors $\mathbf{C} = [\mathbf{A} \mid \mathbf{B}]$, for which the Cohen-Grossberg theorem applies. The subsumption of the unidirectional version of Eqs. (16) and (17) by fixed-weight competitive networks is discussed in Ref. 16.

The Cohen-Grossberg theorem is further extended in the next section when we prove the stability of adaptive BAMs. For simplicity we shall continue to analyze only the additive model, which subsumes the symmetric unidirectional autoassociative circuit model put forth by Hopfield[22] when $M = M^T$.

As shown by Kosko,[10,11] the appropriate bounded Lyapunov or energy function E for the additive BAM system [Eqs. (14) and (15)] is

$$E(A,B) = \sum_i \int_0^{a_i} S'(x_i) x_i dx_i - \sum_i \sum_j S(a_i)\, S(b_j)\, m_{ij} - \sum_i S(a_i) I_i + \sum_j \int_0^{b_j} S'(y_j) y_j dy_j - \sum_j S(b_j)\, J_j. \quad (18)$$

The time derivative of E is computed term by term. The objective is to factor out $S'(a_i)\,\dot{a}_i$ from terms involving inputs to a_i and $S'(b_j)\,\dot{b}_j$ from terms involving inputs to b_j, regroup, then substitute in the STM Eqs. (15) and (16). The time derivative of the integrals is equivalent to the sum of the time derivative of $F[a_i(t)]$ for F_A terms, of $G[b_j(t)]$ for F_B terms. The chain rule gives $dF/dt = dF/da_i\; da_i/dt = S'(a_i)\,\dot{a}_i\, a_i$. The F_A input term gives $S'(a_i)\,\dot{a}_i\, I_i$. The product rule of differentiation is used to compute the time derivative of the quadratic form, which gives the sum of the two endogenous feedback terms in Eqs. (14) and (15) modulated by the respective terms $S'(a_i)\,\dot{a}_i$ and $S'(b_j)\,\dot{b}_j$. Rearrangement then gives

$$\begin{aligned} \dot{E} &= -\sum_i S'(a_i)\dot{a}_i \left[-a_i + \sum_j S(b_j) m_{ij} + I_i \right] \\ &\quad - \sum_j S'(b_j)\dot{b}_j \left[-b_j + \sum_i S(a_i) m_{ij} + J_j \right] \\ &= -\sum_i S'(a_i)\dot{a}_i^2 - \sum_j S'(b_j)\dot{b}_j^2 \\ &\le 0, \end{aligned} \quad (19)$$

on substituting Eqs. (14) and (15) for the terms in brackets. Since $S' > 0$, Eq. (19) implies that $\dot{E} = 0$ if and only if $\dot{a}_i = \dot{b}_j = 0$ for all i and j. At equilibrium all activations and signals are constant. Since M was an arbitrary $n \times p$ real matrix, this proves that every matrix is continuously bidirectionally stable.

As Hopfield[22] has noted, in the high-gain case when the sigmoid signal function S is steep, the integral terms vanish from Eq. (18). Then the equilibria of the continuous energy E in Eq. (18) are the same as those of the bivalent energy E in Eq. (5), namely, the vertices of the product unit hypercube $I^n \times I^p$ or, equivalently, the binary points in $B^n \times B^p$. Continuous BAM convergence then has an intuitive fuzzy set interpretation. A fuzzy set is simply a point in the unit hypercube I^n or I^p. Each component of the fuzzy set is a fit[14] (rather than bit) value, indicating the degree to which that element fits in or belongs to the subset. In a unit hypercube, the midpoint of the hypercube, M = (1/2,1/2, . . . ,1/2) has maximum fuzzy entropy[14] and binary vertices have minimum fuzzy entropy. In a continuous BAM the trajectory of an initial input pattern—an ambiguous or fuzzy key vector—is from somewhere inside $I^n \times I^p$ to the nearest product-space binary vertex. Hence this disambiguation process is precisely the minimization of fuzzy entropy.[11,14]

V. Adaptive BAMs

BAM convergence is quick and robust when M is constant. Any connection topology always rapidly produces a stable contrast-enhanced STM reverberation across F_A and F_B. This stable STM reverberation is not achieved with a lateral inhibition or competitive[12,23] connection topology within the F_A and F_B fields, as it is in the adaptive resonance model,[4] since there are no connections within F_A and F_B. The idea behind an adaptive BAM is to gradually let some of this stable STM reverberation seep into the LTM connections M. Since the BAM rapidly converges and since the STM variables a_i and b_j change faster than the LTM variables m_{ij} change in learning, it seems reasonable that some type of convergence should occur if the m_{ij} change gradually relative to a_i and b_j. Such convergence depends on the choice of learning law for m_{ij}.

In this section we show that, if m_{ij} adapts according to a generalized Hebbian learning law, every BAM adaptively resonates in the sense that all nodes (STM traces) and edges (LTM traces) quickly equilibrate. This real-time learning result extends the Lyapunov approach to the product space $I^n \times I^p \times R^{n \times p}$. The LTM traces m_{ij} tend to learn the associations (A_i,B_i) in unsupervised fashion simply by presenting A_i to the bottom-up field of nodes F_A and simultaneously presenting B_i to the top-down field of nodes F_B. Input patterns sculpt their own attractor basins in which to reverberate. In addition to simple heteroassociative storage and recall, simulation results show that a pure bivalent association (A_i,B_i) can be quickly learned, or abstracted from, noisy gray-scale samples of (A_i,B_i). Many continuous mappings, such as rotation mappings, can also be learned by sampling instantiations of the mappings, often more instantiations than permitted by the storage capacity constraint $m < \min(n,p)$ for simple heteroassociative storage.

How should a BAM learn? How should synapse m_{ij} change with time given successive experience? In the simplest case no learning occurs, so m_{ij} should decay to 0. Passive decay is most simply a model with a first-order decay law:

$$\dot{m}_{ij} = -m_{ij}, \quad (20)$$

so that $m_{ij}(t) = m_{ij}(0)\, e^{-t} \to 0$ as time increases. This simple model contains two ubiquitous features of unsupervised real-time learning models: exponentiation and locality. The mechanism of real-time behavior is exponential modulation. Learning only depends on locally available information, in this case m_{ij}. These two properties facilitate hardware instantiation and increase biological plausibility.

What other information is locally available to the synapse m_{ij}? Only information about a_i and b_j. What

is the simplest way to additively include information about a_i and b_j into Eq. (20)? Multiply or add a_i and b_j—$a_i\ b_j$ or $a_i + b_j$. Multiplicative combination is conjunctive; learning requires signals from both neurons. Additive combination is disjunctive; learning only requires signals from one neuron. Hence associative learning favors the product $a_i\ b_j$. This choice is also an approximation of the correlation coding scheme (9) and produces a naive Hebbian learning law:

$$\dot{m}_{ij} = -m_{ij} + a_i b_j. \tag{21}$$

Again scale constants can be added as desired. Integration of Eq. (21) shows that, in principle, m_{ij} can be unbounded since a_i and b_j can, in principle, just grow and grow. This possibility is sure to occur in feedback networks. So Eq. (21) is unacceptable. Moreover, on closer examination of m_{ij}, which symmetrically connects the ith neuron in F_A with the jth neuron in F_B, we see that the activations a_i and b_j are not locally available to m_{ij}.

Only the signals $S(a_i)$ and $S(b_j)$ are locally available to m_{ij}. In Eq. (8) the bipolar vectors can be interpreted as vectors of threshold signals. So the simplest way to include the locally available information to m_{ij} is to add the bounded signal correlation term $S(a_i)\ S(b_j)$ to Eq. (20). We call this a signal Hebb law:

$$\dot{m}_{ij} = -m_{ij} + S(a_i)S(b_j). \tag{22}$$

Clark Guest (personal communication) notes that (22) is equivalent to the dynamic beam coupling equation in adaptive volume holography. The dynamic system of Eqs. (16), (17), and (22) defines an adaptive BAM. Suppose all nodes and edges have equilibrated. Then the equilibrium value of m_{ij} is found by setting the right-hand side of Eq. (22) equal to 0:

$$m_{ij} = S_e(a_i)S_e(b_j). \tag{23}$$

The signal Hebb law is bounded since the signals are bounded. Suppose for definiteness that S is a bipolar signal function. Then

$$-1 \le S(a_i)S(b_j) \le 1. \tag{24}$$

The signal product is +1 if both signals are +1 or both are −1. The product is −1 if one signal is +1 and the other is −1. Thus the signal product behaves as a biconditional or equivalence operator in a fuzzy or continuous-valued logic. This biconditionality underlies the interpretation of the association (A_i,B_i) as the conjunction IF A_i THEN B_i, and IF B_i THEN A_i. Moreover, the bipolar endpoints −1 and +1 can be expected to abound with a steep bounded S.

Suppose m_{ij} is maximally increasing due to $S(a_i)\ S(b_j) = 1$. Then Eq. (22) reduces to the simple first-order equation

$$\dot{m}_{ij} + m_{ij} = 1, \tag{25}$$

which integrates to

$$\begin{aligned} m_{ij}(t) &= e^{-t}m_{ij}(0) + \int_0^t e^{(s-t)}ds \\ &= e^{-t}m_{ij}(0) + (1 - e^{-t}) \\ &\to 1 \text{ as } t \text{ increases for any initial } m_{ij}(0). \end{aligned} \tag{26}$$

Similarly, if m_{ij} is maximally decreasing, the right-hand side of Eq. (24) is −1 and m_{ij} approaches +1 exponentially fast independent of initial conditions. This agrees with Eq. (23). The signal Hebb law (22) asymptotically approaches the bipolar correlation learning scheme (8) for a single data pair. So the learning BAM for simple heteroassociative storage can still be expected to be capacity constrained by $m < \min(n,p)$.

The BAM memory medium produced by Eq. (22) is almost perfectly plastic. Scaling constants in Eq. (22) must be carefully chosen. In particular, the forget term $-m_{ij}$ in Eq. (22) must be scaled with a constant less than unity. Otherwise present learning washes away past learning $m_{ij}(0)$. In practice this means that a training list of associations $(A_1,B_1), \ldots, (A_m,B_m)$ should be presented to the adaptive BAM system more than once if each pair (A_i,B_i) is presented for the same length of time. Alternatively, the training list can be presented once if the first pair (A_1,B_1) is presented longer than (A_2,B_2) is presented, (A_2,B_2) longer than (A_3,B_3), (A_3,B_3) longer than (A_4,B_4), and so on. This holds because the general integral solution to Eq. (22) is an exponentially weighted average of sampled patterns.

In what sense does the adaptive BAM converge? We prove below that it always converges in the sense that nodes and edges rapidly equilibrate or resonate when environmentally perturbed. Recall and learning can simultaneously occur in a type of adaptive resonance.[4–9]

At this point it is instructive to distinguish simple adaptive BAM behavior from standard adaptive resonance theory (ART) behavior. The high-level processing behavior of the Carpenter-Grossberg[4] ART model can be sketched as follows. Only one node in F_B fires at a time, the instar[8] node b_j that won the competition for bottom-up activation when a binary input pattern was presented to F_A. The winner b_j then fans out its spatial pattern or outstar[8] to the nodes in F_A. If this fan-out pattern sufficiently matches the input pattern presented to F_A, a stable pattern of STM reverberation is set up between F_A and F_B, learning can occur (but need not), and instar b_j has recognized or categorized the input pattern. Otherwise b_j is shut off and another instar winner b_k fans out its spatial pattern, etc., until a match occurs or, if no match occurs, until the binary input pattern trains some uncommitted node b_u to be its instar. Hence each instar node b_j in the ART model recognizes or categorizes a single input pattern or set of input patterns, depending on how high a degree of match is desired. Match degree can be deliberately controlled. Direct access to a trained instar is assured only if the input matches exactly, or nearly, the pattern learned by the instar. The more novel the pattern presented to F_A, and the higher the desired degree of match, the longer the ART system tends to search its instars to classify it.

In the adaptive BAM every F_B node b_j in parallel fans out its outstar across F_A when a STM pattern is active across F_A. The signal Hebb law (22) distributes

recognition capability across all the edges of all the b_j nodes so that most bivalent associations are unaffected by removing a particular node. The closest analog to a specifiable degree of match in a BAM is the storage-capacity relationship between pattern number and pattern dimensionality, $m < \min(n,p)$. The closer m is to the maximum reliable capacity, the greater the match, between an input pattern and a stored association (A_i,B_i), required to evoke (A_i,B_i) into a stable STM reverberation. When m is small relative to the maximum capacity, there tend to be few basins of attractions in the state space $I^n \times I^p$, the basins tend to have wide diameters, and they tend to correspond to the stored associations (A_i,B_i). Each stored association tends to recognize or categorize a large set of input stimuli. When m is large, there tend to be several basins, with small diameters. When m is large enough, only the exact patterns A_i or B_i will evoke (A_i,B_i). Within capacity constraints, all inputs tend to fall into the basin of the nearest stored association and thus have direct access to nearest stored associations. Novel patterns are classified or misclassified as rapidly as more familiar patterns.

Learning can also occur in an adaptive BAM during the rapid recall process. Familiar patterns tend to strengthen or restrengthen the reverberating associations they elicit. Novel patterns tend to misclassify to spurious energy wells (attractor basins), which in effect recognize them, or by Eq. (22) they tend to dig their own energy wells, which thereafter recognize them. As the simulation results discussed below show, many more patterns can be stably presented to the BAM than $\min(n,p)$ if they resemble stored associations. Otherwise the forgetting effects of Eq. (22) prevail and at any moment the adaptive BAM tends to remember no more than the most recent $\min(n,p)$-many distinct inputs (elicited associations).

We now prove that the adaptive BAM converges to local energy minima. Denote the bounded energy function in Eq. (18) by F. Then the appropriate energy or Lyapunov function for the adaptive BAM dynamic system of Eqs. (16), (17), and (22) is simply

$$E(\mathrm{A},\mathrm{B},\mathrm{M}) = F + 1/2 \sum_i \sum_j m_{ij}^2, \tag{27}$$

since the time derivative of $1/2\, m_{ij}^2$ is $m_{ij}\, \dot{m}_{ij}$. This new energy function is bounded since each m_{ij} is bounded. When the product rule of differentiation is applied to the time-varying triple product in the quadratic form component of F [Eq. (18)], we get the triple sum

$$\dot{m}_{ij} S(a_i) S(b_j) + S'(a_i) \dot{a}_i m_{ij} S(b_j) + S'(b_j) \dot{b}_j m_{ij} S(a_i).$$

In the nonlearning continuous BAM the first term of this triple sum was zero and the new sum of squares in Eq. (27) was constant and hence made no contribution to Eq. (19). Now the time derivative of E in Eq. (27) gives, on rearrangement,

$$\begin{aligned} \dot{E} &= -\sum_i \sum_j \dot{m}_{ij} [S(a_i) S(b_j) - m_{ij}] - \sum_i S' \dot{a}_i^2 - \sum_j S' \dot{b}_j^2 \\ &= -\sum_i \sum_j \dot{m}_{ij}^2 - \sum_i S'(a_i) \dot{a}_i^2 - \sum_j S'(b_j) \dot{b}_j^2 \\ &\le 0, \end{aligned} \tag{28}$$

on substituting the signal Hebb learning law (22) for the term in brackets in Eq. (28). Hence an adaptive BAM is a dissipative dynamic system that generalizes the nonlearning continuous BAM dissipative system. When energy stability is reached, when $\dot{E} = 0$, Eq. (28) and $S' > 0$ imply that both edges and nodes have stabilized: $\dot{m}_{ij} = \dot{a}_i = \dot{b}_j = 0$ for all i and j. Hence every signal Hebb BAM adaptively resonates. This result further generalizes in a straightforward way to any number of layered BAM fields that are interconnected, not necessarily contiguously, by Eq. (22).

Can an adaptive BAM learn and recall simultaneously? In the ART model[4] a mechanism of attentional gain control [inhibition due to the sum of F_B signals $S(b_j)$] is introduced to enable neurons a_i in F_A to distinguish environmental inputs I from top-down feedback patterns B. In principle, an attentional gain control mechanism can also be added to an adaptive BAM. Short of this new mechanism, how can neuron a_i distinguish external input I_i and internal feedback input from F_B? In Eq. (14) these terms both additively effect the time change of a_i. So external and internal feedback to a_i can only differ in their patterns of magnitude and duration over some short time interval. If the magnitude and duration of inputs are indistinguishable, the inputs are indistinguishable to a_i. When they differ, a_i can in principle learn and recall simultaneously.

Suppose a randomly fluctuating, uninformative environment confronts the adaptive BAM. Then I_i tends to have zero mean in short time intervals. This allows a_i to be driven by internal feedback from F_B. If learning is permitted, familiar STM reverberations, evoked perhaps by other a_k (or b_j), can be strengthened. When I_i remains relatively constant over an interval, a new pattern can be learned, and can be learned while F_A and F_B reverberate, eventually dominating those reverberations. If the reverberations are spurious, learning is enhanced by appropriately weighting I_i. In simulations, scaling I_i by p, the number of neurons in F_B, has proved effective presumably because it balances the magnitude of I_i against the magnitude of the internal F_B feedback sum in Eq. (14).

An extension of these ideas is the sampling adaptive BAM. There is a trade-off between learning time and learning samples. The standard learning model is to present relatively few samples for long lengths of learning time, typically until learning converges or is otherwise terminated, as in simple heteroassociative storage, or to present few samples over and over, as in backpropagation.[17-20] In what we shall call sampling learning several samples are presented briefly—typically many more patterns than neuron dimensionality—and the underlying patterns, associations, or mappings are better learned as sample size increases. Learning is not allowed to converge. Only a brief pulse of learning occurs for each sample. When the sam-

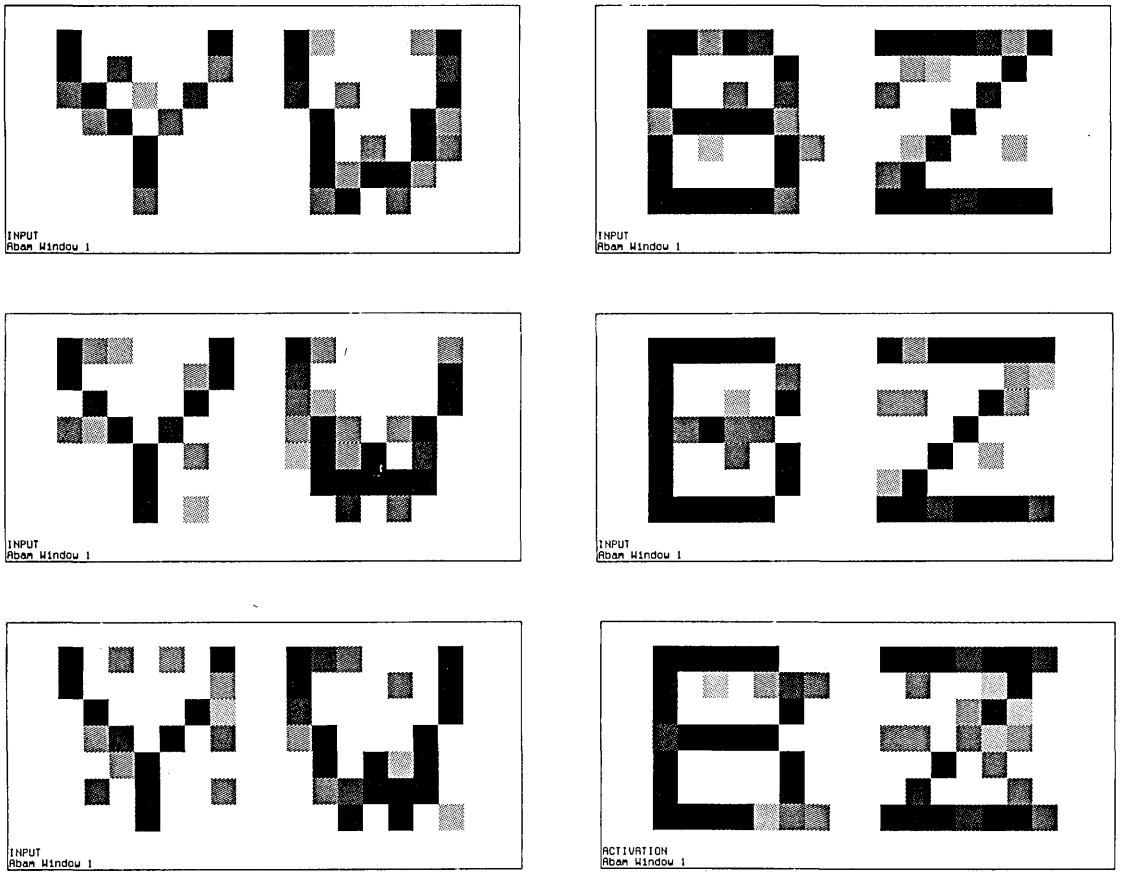


Fig. 4. Sampling adaptive BAM noisy training set. Forty-eight randomly generated gray-scale noise patterns are presented to the system. Unlike in simple heteroassociative storage, no sample is presented long enough for learning to fully or nearly converge. Twenty-four of the samples are noisy versions of the bipolar association (Y,W); twenty-four are noisy versions of (B,Z). Three samples are displayed from each training set. Samples are presented four at a time—from the (Y,W) training set, then four from the (B,Z) training set, then the next four from the (Y,W) training set, etc. Both fields F_A and F_B contain forty-nine samples, violate the storage capacity $m \ll \min(n,p)$ for simple heteroassociative storage.

pling learning technique is applied to the adaptive BAM, a sampling adaptive BAM results. For example, an adaptive BAM can rapidly learn a rotation mapping, if $n = p$, by simply presenting a few spatial patterns at F_A and concurrently presenting the same pattern rotated some fixed degree at F_B. Thereafter any pattern presented at F_A produces the stable STM reverberation with the input pattern at F_A and its rotated version at F_B.

We note that Hecht-Nielsen[24] has developed his feedforward counterpropagation sampling learning technique for learning continuous mappings, and probability density functions that generate mappings, by applying Grossberg's outstar learning theorem[8,9] and by applying the sampling learning technique to Grossberg's unsupervised competitive learning[2,23]:

$$\dot{m}_{ij} = (i_i - m_{ij})b_j, \tag{29}$$

which is also used in the ART model,[4] where $(i_1, \ldots, i_n)$ is a normalized input pattern or probability distribution presented to F_A and b_j provides competitive modulation, e.g., $b_j = 1$ if b_j wins the F_B instar competition for activation and $b_j = 0$ otherwise. For simple autoassociative storage the competitive instar learning law (29) is also dimension bounded for nonsampling learning. No more distributions at F_A can be recognized at F_B than, obviously, the number p of instar nodes at F_B. Yet Hecht-Nielsen[24] has demonstrated that sampling learning with Eq. (29) can learn a sine wave, which has minimal dimensionality, well with thirty neurons and a few hundred random samples, almost perfectly with a few thousand random samples.

Figures 4–6 display the results of a sampling BAM experiment. F_A and F_B each contain forty-nine gray-scale neurons arranged in a 7×7 pixel tray. The output of the bipolar logistic signal function $S(x)$ is discretized to six gray-scale levels, where $S(x) = -1$ is white and $S(x) = 1$ is black. $S(x) = -1$ if activation $x < -51$, $S(x) = 1$ if $x > 51$. Forty-eight randomly generated gray-scale noise patterns are presented to the adaptive BAM. The forty-eight samples violate the storage capacity $m \ll \min(n,p)$ for simple heteroassociative storage. Figure 4 displays six of

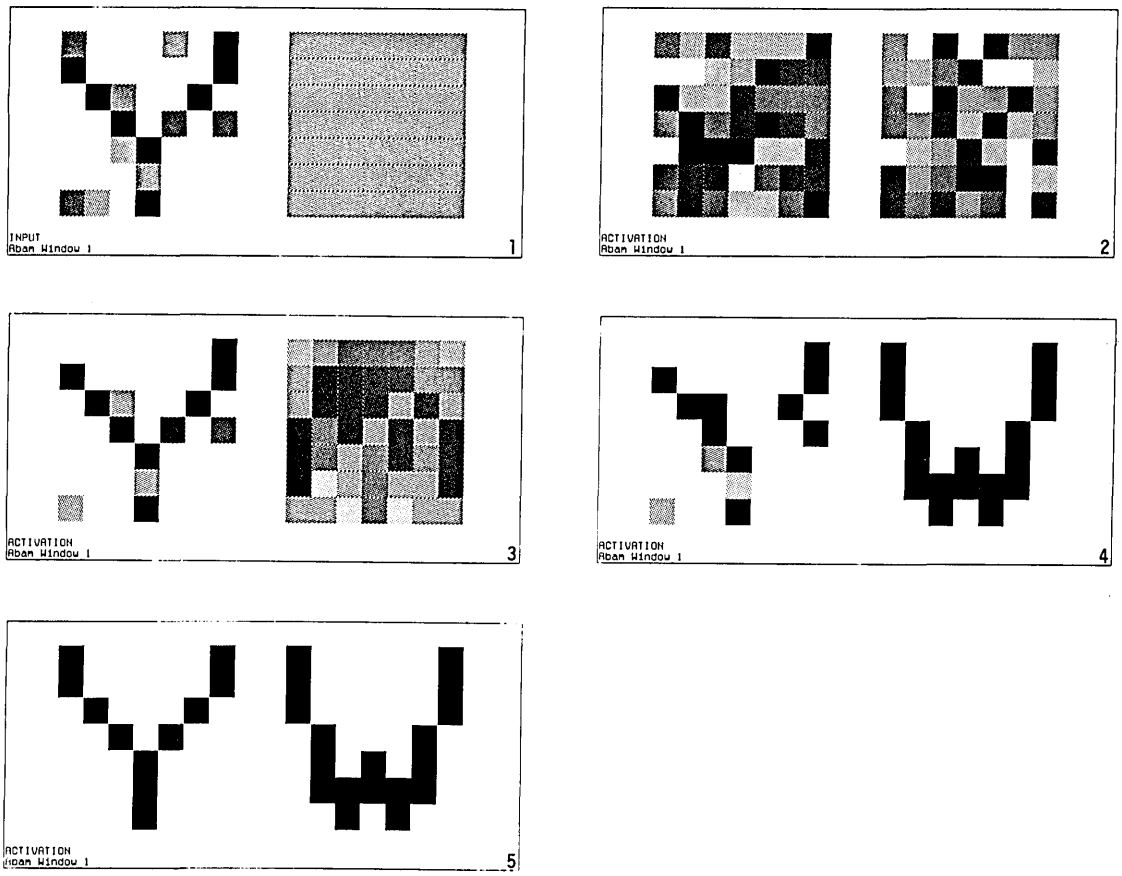


Fig. 5. Sampling adaptive BAM associative recall and abstraction. A new noisy version of Y is presented to field F_A. Initial BAM STM activation across F_A and F_B is random. The BAM converges to the pure bipolar association (Y,W) it has never experienced but has abstracted from the noisy training samples in Fig. 4.

these random samples. Twenty-four of the samples are noisy versions of the bipolar association (Y,W); twenty-four are noisy versions of (B,Z). Noise was created by picking numbers in $[-60,60]$ according to a uniform distribution, then adding them to the activation values, -52 or 52, underlying the bivalent signal values making up (Y,W) and (B,Z). Unlike in simple heteroassociative storage, no sample is presented long enough for learning to fully or nearly converge. Samples are briefly presented four at a time—four from the (Y,W) training set, then four from the (B,Z) training set, then the next four from the (Y,W) training set, and so on to exploit the exponentially weighted averaging effects of the signal Hebb learning law (22).

Figure 5 demonstrates recall and abstraction with the sampling adaptive BAM. A new noisy version of Y is presented to field F_A. The initial STM activation across F_A and F_B is random. The BAM converges to the pure bipolar association (Y,W) it has never experienced but has abstracted from the noisy training samples. As in Plato's theory of ideals—and unlike the naive empiricist denial of abstraction of Locke, Berkeley, and Hume—it is as if the BAM learns redness from red things, smoothness from smooth things, triangularity from triangles, etc., and thereafter associates new red things with redness, not with most-similar old red things.

In Figure 6 the BAM is thinking about the STM reverberation (Y,W). A new noisy version of Z is presented to field F_B, superimposing it on the (Y,W) reverberation. The reverberating thought is soon crowded out of STM by the environmental stimulus Z. The BAM again converges to the unobserved pure bipolar association, this time (B,Z), it abstracted from the noisy training samples.

This research was supported by the Air Force Office of Scientific Research (AFOSR F49620-86-C-0070) and the Advanced Research Projects Agency of the Department of Defense under ARPA Order 5794. The author thanks Robert Sasseen for developing all software and graphics.

References

1. T. Kohonen, "Correlation Matrix Memories," IEEE Trans. Comput. C-21, 353 (1972).

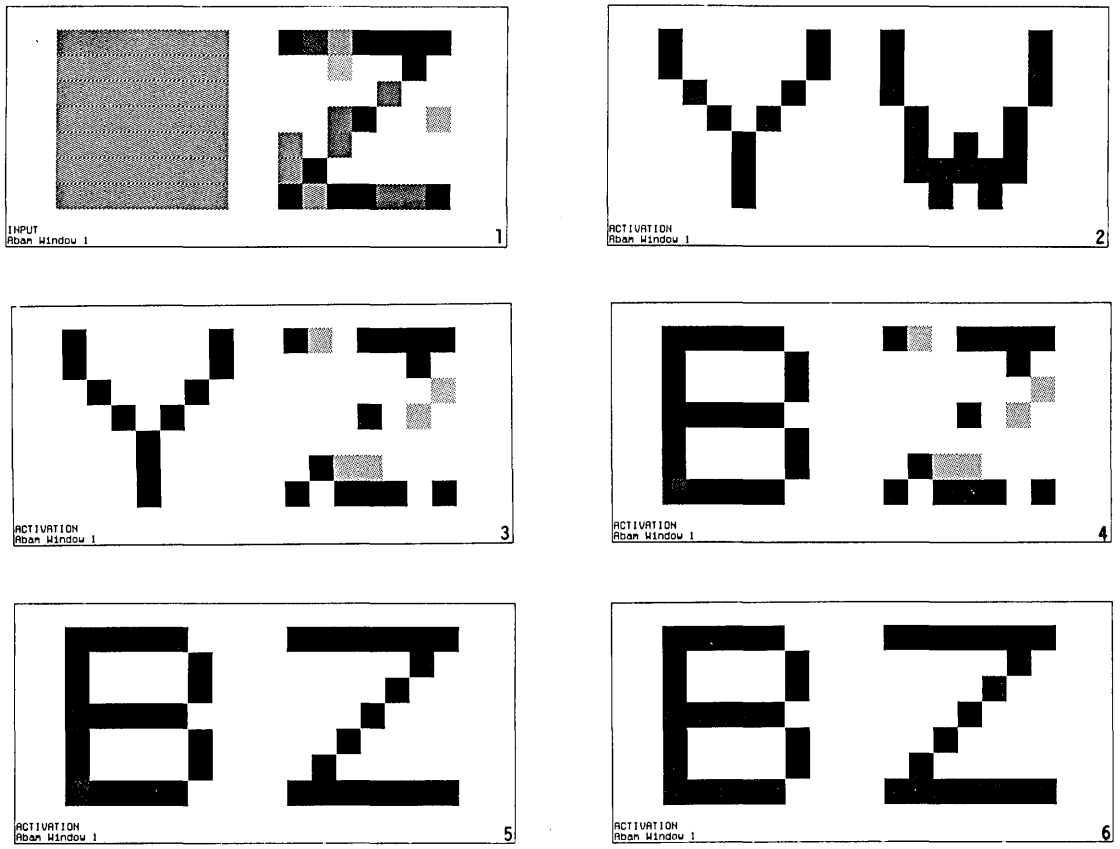


Fig. 6. Sampling adaptive BAM STM superimposition and associative recall. A new noisy version of Z is presented to field F_B. This time the bipolar association (Y,W) recalled in Fig. 5 is reverberating in STM. This thought is soon crowded out of STM by the environmental stimulus Z. Again the BAM converges to the unobserved pure bipolar association, this time (B,Z), it abstracted from the noisy training samples.

2. T. Kohonen, *Self-Organization and Associative Memory* (Springer-Verlag, New York, 1984).
3. J. A. Anderson, J. W. Silverstein, S. A. Ritz, and R. S. Jones, "Distinctive Features, Categorical Perception, and Probability Learning: Some Applications of a Neural Model," Psychol. Rev. **84,** 413 (1977).
4. G. A. Carpenter and S. Grossberg, "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine," Comput. Vision Graphics Image Process. **37,** 54 (1987).
5. S. Grossberg, "Adaptive Pattern Classification and Universal Recoding, II: Feedback, Expectation, Olfaction, and Illusions," Biol. Cybern. **23,** 187 (1976).
6. S. Grossberg, "A Theory of Human Memory: Self-Organization and Performance of Sensory-Motor Codes, Maps, and Plans," Prog. Theor. Biol. **5,** 000 (1978).
7. S. Grossberg, "How Does a Brain Build a Cognitive Code?," Psychol. Rev. **87,** 1 (1980).
8. S. Grossberg, *Studies of Mind and Brain: Neural Principles of Learning, Perception, Development, Cognition, and Motor Control* (Reidel, Boston, 1982).
9. S. Grossberg, *The Adaptive Brain, I and II* (North-Holland, Amsterdam, 1987).
10. B. Kosko, "Bidirectional Associative Memories," IEEE Trans. Syst. Man Cybern. **SMC-00,** 000 (1987).
11. B. Kosko, "Fuzzy Associative Memories," in *Fuzzy Expert Systems*, A. Kandel, Ed. (Addison-Wesley, Reading, MA, 1987).
12. S. Grossberg, "Contour Enhancement, Short Term Memory, and Constancies in Reverberating Neural Networks," Stud. Appl. Math. **52,** 217 (1973).
13. W. S. McCulloch and W. Pitts, "A Logical Calculus of the Ideas Immanent in Nervous Activity," Bull. Math. Biophys. **5,** 115 (1943).
14. B. Kosko, "Fuzzy Entropy and Conditioning," Inf. Sci. **40,** 165 (1986).
15. J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," Proc. Natl. Acad. Sci. U.S.A. **79,** 2554 (1982).
16. M. A. Cohen and S. Grossberg, "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks," IEEE Trans. Syst. Man Cybern. **SMC-13,** 815 (1983).
17. D. B. Parker, "Learning Logic," Invention Report S81-64, File 1, Office of Technology Licensing, Stanford U. (Oct. 1982).
18. D. B. Parker, "Learning Logic," TR-47, Center for Computational Research in Economics and Management Science, MIT (Apr. 1985).
19. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," ICS Report 8506, Institute for Cognitive Science, U. California San Diego (Sept. 1985).
20. P. J. Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences," Ph.D. Dissertation in Statistics, Harvard U. (Aug. 1974).
21. B. Kosko and C. Guest, "Optical Bidirectional Associative Me-

mories," Proc. Soc. Photo-Opt. Instrum. Eng. 758, (1987).
22. J. J. Hopfield, "Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons," Proc. Natl. Acad. Sci. U.S.A. 81, 3088 (1984).
23. S. Grossberg, "Adaptive Pattern Classification and Universal Recoding, I: Parallel Development and Coding of Neural Feature Detectors," Biol. Cybern. 23, 121 (1976).
24. R. Hecht-Nielsen, "CounterPropagation Networks," in *Proceedings, First International Conference on Neural Networks* (IEEE, New York, 1987).

Patter *continued from page 4946*

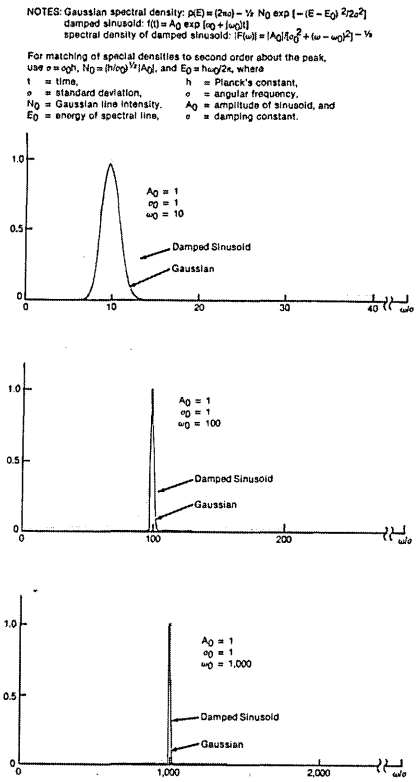


Fig. 2. Spectral density of a damped sinusoid and a Gaussian approximate each other near the peak when the parameters of the two distributions are chosen appropriately.

no intervention by the operator is required: It is not necessary to exercise subjective judgment to set parameters that depend on the type of spectrum being analyzed. The technique is based on the similarities between the zero- and second-order terms of the Taylor-series expansions of a Gaussian distribution and of a damped sinusoid (the first-order terms are zero): The two zero-order terms and the two second-order terms become equal when the peak amplitudes of the two distributions are set equal at the sinusoidal-oscillation frequency, and the standard deviation of the Gaussian is matched, on the frequency scale, to the damping constant of the sinusoid. Thus, the two distributions are made to approximate each other in the vicinity of the peak (see figure). However, the two distributions differ far from the peak, where the damped sinusoid is more consistent with measured data.

A principal advantage of the algorithm is that there is no requirement to adjust weighting factors or other parameters when analyzing general x-ray spectra. Thus, there is no erroneous subjective bias. All spectra, no matter how complicated, are analyzed with the same routine. The algorithm uses only the magnitude Fourier spectrum to calculate the distribution parameters, whereas some analytical procedures require the use of complex (phase and magnitude) spectral data. In this case, only the magnitude data are available from the spectral densities.

This work was done by David Nicolas, Clayborne Taylor, and Thomas Wade of Marshall Space Flight Center. Inquiries concerning rights for the commercial use of this invention should be addressed to the Patent Counsel, Marshall Space Flight Center. Refer to MFS-26039.

Baseband processor for communication satellites

A baseband processing (BBP) system for advanced satellite communications has been successfully demonstrated. This system provides increased data capacity through frequency-reusing multibeam antenna systems, using time-division multiple access (TDMA) and onboard satellite switching. Large numbers of thin-route trunking stations and user-based earth terminals are handled efficiently by satellite baseboard switching. The baseband processor that performs this function is one of the primary subsystems for the next generation of satellite communication systems. With the BBP system, the satellite can route data messages individually among locations anywhere in the continental United States. The function of the BBP system as a part of the satellite transponder system is to process, control, and route message traffic among individual users equipped with onsite ground terminals and among thin-route trunking terminals served by both the scanning-beam and fixed-beam antennas. The BBP system is required to include the nonblocking switching of data on individual channels, the interconnection of any terminals to any other terminal on a point-to-point basis, and interconnection of any terminal to any other set of terminals in a limited broadcast mode.

A description of the operation of the baseband processor begins with the incoming uplink traffic. Messages are transmitted to the satellite in a TDMA format. The messages arriving at the input of the BBP system are demodulated down to baseband. Messages that may have been encoded for rain-fade compensation are decoded. The messages are then stored in one of two input memories. While one memory is being loaded, the other is being unloaded. The 64-bit-word messages being unloaded from the input memories are routed to the output memories through the routing switch on a word-by-word basis. The output memories work like the input memories: one is unloading while the other is being loaded. The message stream of 64-bit words being unloaded from the memories is encoded where needed for rain-fade compensation and modulated to the downlink frequency, whereupon it exits the BBP system. The BBP-system digital routing controller, as programed from the master-control ground station, routes the messages properly through the BBP system, and controls the uplink and downlink scanning-beam sequencing.

The user, although his/her messages are being broken down, compressed into bursts, and time-division multiplexed with many other messages, sees a continuous, unbroken connection with whomever he/she is communicating. Figure 3 shows the functional diagram for

continued on page 4971