

Unsupervised Learning in Noise

BART KOSKO, MEMBER, IEEE

Abstract—The structural stability of real-time unsupervised learning in feedback dynamical systems is demonstrated with the stochastic calculus. Structural stability allows globally stable feedback systems to be perturbed without changing their qualitative equilibrium behavior. These stochastic dynamical systems are called *random adaptive bidirectional associative memory* (RABAM) models, which include several popular nonadaptive and adaptive feedback models, such as the Hopfield circuit and the ART-2 model. RABAM networks can adapt with different stable unsupervised learning laws. These include the signal Hebb, competitive, and differential Hebb laws. A new hybrid learning law, the *differential competitive law*, which uses the neuronal signal velocity as a local unsupervised reinforcement mechanism, is introduced and its coding and stability behavior in feedforward and feedback networks is examined. This analysis is facilitated by the recent Gluck-Parker pulse-coding interpretation of signal functions in differential Hebbian learning systems. The second-order behavior of RABAM Brownian-diffusion systems is summarized by the RABAM noise suppression theorem: The mean-squared activation and synaptic velocities decrease exponentially quickly to their lower bounds, the instantaneous noise variances driving the system. This result is extended to the RABAM annealing model, which provides a unified framework from which to analyze Geman-Hwang combinatorial optimization dynamical systems and continuous Boltzmann machine learning.

I. STRUCTURAL STABILITY IN HARDWARE, BIOLOGY, AND MANIFOLDS

HOW robust are unsupervised learning systems? What happens if real-time synaptic mechanisms are perturbed in real time? Will shaking disturb or prevent equilibria? What effect will thermal noise processes, electromagnetic interactions, and component malfunctions have on large-scale implementations of unsupervised neural networks? How biologically accurate are unsupervised neural models that do not model the myriad electrochemical, molecular, and other processes found at synaptic junctions and membrane potential sites?

These questions are different ways of asking a more general question: is unsupervised learning *structurally stable*? Structural stability [9], [42] allows globally stable feedback systems to be perturbed without changing their qualitative equilibrium behavior. This increases the reliability of large-scale hardware implementations of such networks. It also increases their biological plausibility,

since the myriad synaptic and neuronal processes missing from neural network models now are modeled, but as net random unmodeled effects that do not affect the structure of the global network computations.

Structural stability differs from the global stability, or convergence to fixed points, that endows some feedback networks with content addressable memory, and other, computational properties. Globally stable systems can be sensitive to initial conditions. Different inputs states can converge to different limit states; else memory capacity is trivial. Structural stability is insensitivity to small perturbations. Such perturbation preserves qualitative properties. In particular, basins of attractions maintain their basic shape. In some intuitive sense, chaos [36] is the antithesis of structural stability, or, more accurately, structurally stable fixed-point attractors (since chaotic attractors can be structurally stable).

The formal approach to structural stability uses the transversality techniques of differential topology [17], the study of global properties of differentiable manifolds. Manifolds A and B have nonempty transversal intersection in R^n if the tangent spaces of A and B span R^n at every point of intersection, if locally the intersection looks like R^n . Two lines intersect transversely in the plane but not in 3-space, 4-space, or higher n -space. If the lines are shaken in 2-space, they still intersect. If shaken in 3-space, the lines may no longer intersect. In Fig. 1, manifolds A and B intersect transversely in the plane at points a and b . Manifolds B and C do not intersect transversely at c .

An indirect approach to structural stability uses the calculus of stochastic differential and integral equations [35], [41]. This is the approach used in this paper. The stochastic-calculus approach abstracts statistically relevant behavior from large sets of functions. The differential-topological approach, in contrast, is concerned with all possible behavior of all functions (open dense sets of functions). This makes the analysis extremely abstract and calculations cumbersome and often impractical.

The stochastic calculus is difficult to work with as well, but usually less difficult than transversality techniques. The new complexity that arises in passing from systems of differential equations to systems of stochastic differential equations is due to the nature of solution points. In algebraic equations, such as $2x + 3 = 4x$, points in the solution space are numbers. Solutions to differential equations are functions. Solutions to stochastic differential equations are random processes [41].

Manuscript received April 10, 1989; revised October 9, 1989. This work was supported by the Air Force Office of Scientific Research (AFOSR-88-0236). An earlier version of this paper was presented at the 1989 International Joint Conference on Neural Networks (IJCNN-89), Washington, DC, June 18-22, 1989.

The author is with the Department of Electrical Engineering, Systems, Signal, and Image Processing Institute, University of Southern California, Los Angeles, CA 90089-0272.

IEEE Log Number 8933088.

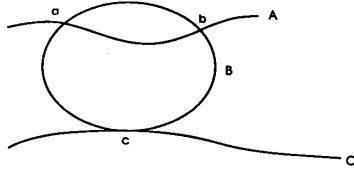


Fig. 1. Manifold intersection in the plane (manifold R^2). Intersection points a and b are transversal. Point c is not: manifolds B and C need not intersect if even slightly perturbed. No points are transversal in 3-space unless B is a sphere.

Below we demonstrate the structural stability of many types of unsupervised learning in the stochastic sense. The key idea is to use the scalar-valued Lyapunov function of globally stable feedback networks but in a random framework. Then the old Lyapunov function is a random variable at each moment of time t , so it cannot be minimized as when it was a scalar at each t . The trick is to minimize its expectation, its average value, which is a scalar at t .

II. FOUR UNSUPERVISED ASSOCIATIVE LEARNING LAWS

The distinction between supervised and unsupervised learning depends on information. In pattern-recognition theory, for instance, the distinction is in terms of knowledge of class boundaries. Pattern recognition is supervised if the training algorithm requires knowing the class membership of the training samples, unsupervised if it does not require it.

A similar distinction holds in neural networks. Supervised learning invariably refers to deliberate gradient descent in the space of all possible synaptic values. Class membership information is needed to compute the numerical error vector or error signal that guides the gradient descent.

Unsupervised learning usually refers to the modification of biological synapses with physically local signal information. Class membership information of training samples is not needed. These systems adaptively cluster patterns into classes by, for example, evolving "winning" neurons in a competition for activation, or by evolving different basins of attraction in the state space. We shall restrict our attention to such biologically motivated learning methods, knowing that other types of unsupervised learning are possible and may be of practical engineering value.

Unsupervised learning laws are first-order differential equations that describe how synapses evolve in time with *locally available* information. This information usually involves synaptic properties or neuronal signal properties. In principle, and in mammalian brains or optoelectronic integrated circuits, other types of information may be locally available for computation, glial cells, specific and nonspecific hormones, background electromagnetic effects, or light pulses. These phenomena are modeled below as net random parameters. For the moment they will be ignored. Locality allows asynchronous synapses to operate in real time. Mathematically, it also greatly shrinks the function space of possible unsupervised learning laws.

Associativity further shrinks the function space. Globally, neural networks associate patterns with patterns. They estimate continuous functions. Locally, synapses are required to associate signals with signals. This leads to conjunctive, or multiplicative, learning laws constrained by locality. This in turn leads to at least three types of learning laws and a new hybrid law.

The four unsupervised associative learning laws discussed in this section are 1) the signal Hebb learning law, 2) the competitive learning law, 3) the differential Hebb learning law, and 4) the new hybrid law, the differential competitive learning law.

A. Signal Hebbian Learning

The *signal Hebb* learning law correlates neuronal signals, not activations:

$$\dot{m}_{ij} = -m_{ij} + S_i^X(x_i) S_j^Y(y_j) \quad (1)$$

where the overdot indicates time differentiation, m_{ij} is the synaptic efficacy of the directed axonal edge from the i th neuron in field F_X to the j th neuron in field F_Y , x_i and y_j are the respective real-valued activations or membrane potentials of the connected neurons, and S_i^X and S_j^Y , hereafter abbreviated to S_i and S_j , are the bounded monotone-nondecreasing signal functions of the connected neurons that transduce their time-averaged potential differences into time-averaged frequencies of pulse trains, and where, as in all equations in this paper, scaling constants can be multiplied or added where desired. The logistic signal function $S(x) = (1 + e^{-cx})^{-1}$, with $c > 0$, remains the most popular signal function for simulations and applications. The logistic signal function is also strictly monotone increasing, since $S' = dS(x)/dx = cS(1 - S) > 0$. Strict monotonicity strengthens stability results.

The solution to (1) is an integral equation since in general x_i and y_j depend on m_{ij} . The key component of this integral equation is an exponentially weighted average of sampled patterns:

$$m_{ij}(t) = m_{ij}(0)e^{-t} + \int_0^t S_i(s) S_j(s) e^{s-t} ds. \quad (2)$$

The exponential weight is inherent in the first-order structure of (1). It produces a *recency effect* on memory, as in our everyday exponential decrease in retained information. This well-known recency effect is the thrust of philosopher David Hume's quote: "The liveliest thought still is inferior to the dullest sensation." Nothing is more vivid than now.

B. Competitive Learning

The *competitive learning law* is obtained from (1) if the passive decay term $-m_{ij}$ is modulated by the appropriate local signal:

$$\dot{m}_{ij} = S_j[S_i - m_{ij}]. \quad (3)$$

The "competitiveness" in (3) is indirect. The assumption is that neurons compete for activation in the field F_Y

in the sense that the symmetric (distance-dependent) intrafield connections of F_Y are laterally inhibitive: the square symmetric matrix Q of intrafield connections is positive main-diagonal and nonpositive off-diagonal, or, more generally, Q has nonnegative blocks on its main diagonal and nonpositive blocks elsewhere. Then S_j is a win-loss index of the j th F_Y neuron's performance. In practice [39] S_j is invariably a 0-1 threshold function or steep logistic function, which behaves as a threshold function. Then (3) says *learn only if win*. If the j th unit wins, the signal pattern $S(X) = (S_1(x_1), \dots, S_n(x_n))$ generated at F_X is encoded as the j th column of the n -by- p connection matrix exponentially quickly. This "grandmother synapse" effect differs from Hebbian learning, where pattern information is superimposed on all of M . Then every synapse participates in learning new patterns while, unfortunately, forgetting learned patterns.

Both (1) and (2) were studied as early as the 1960's by Grossberg [12]. Kohonen [24] and Hecht-Nielsen [15] use the competitive law (3) statistically for unsupervised clustering in their respective self-organizing map and counterpropagation networks. The p columns of M then tend toward the centroids of the sampled p decision classes, even though the underlying probability density functions are unknown.

C. Differential Hebbian Learning

The *differential Hebb* law [25]–[27], [32], [33], and its variants, correlates signal velocities as well as signals:

$$\dot{m}_{ij} = -m_{ij} + S_i S_j + \dot{S}_i \dot{S}_j \quad (4)$$

where, by the chain rule

$$\frac{dS_i(x_i)}{dt} = \frac{dS_i}{dx_i} \frac{dx_i}{dt} = S'_i \dot{x}_i.$$

If signals are locally available to synapses, so are signal velocities, at least implicitly. Since the signal function S_i is an abstraction of time-averaged spiking frequencies, S_i is often assumed nonnegative. Then Hebbian synapses (1) can only grow in time. Signal velocities, of course, can be both positive and negative. Correlated (lagged) signals provide a local "arrow of time" that synapses can exploit [33] to encode time-varying patterns as limit cycles. Klopff [21]–[23] independently arrived at a similar discrete (difference) version of (4) in his drive-reinforcement theory of animal learning.

Recently Gluck and Parker [10], [11] showed that differential Hebbian learning becomes significantly more plausible in nervous systems if we recall that real neurons transmit discrete pulse-coded information and we structure the signal functions S_i and S_j accordingly. Suppose x_i and y_j are pulse functions: $x_i(t) = 1$ if a pulse occurs at time t , 0 if not, and similarly for $y_j(t)$. Then the signal frequencies S_i and S_j can be estimated as exponentially weighted time averages:

$$S_i(t) = \int_{-\infty}^t x_i(s) e^{s-t} ds \quad (5)$$

$$S_j(t) = \int_{-\infty}^t y_j(s) e^{s-t} ds. \quad (6)$$

By recalling the form of the solution to a linear inhomogeneous, first-order differential question, the signal velocities are seen to be simple, locally available, differences:

$$\dot{S}_i(t) = x_i(t) - S_i(t) \quad (7)$$

$$\dot{S}_j(t) = y_j(t) - S_j(t). \quad (8)$$

Thus a signal velocity has the form of a reinforcement signal: a pulse less the current expected frequency of pulses. As Gluck and Parker observe, not only are these differences locally available, they can be computed in real time without unstable differencing techniques.

For stability purposes, we note another consequence of pulse-coded signal functions. They show how Hebbian learning can be a special case of differential Hebbian learning. Suppose the Hebb product $S_i S_j$ in (4) is scaled down to zero:

$$\dot{m}_{ij} = -m_{ij} + \dot{S}_i \dot{S}_j. \quad (9)$$

This is the "classical" differential Hebb law [25]–[27]. Then substituting (7) and (8) into (9) gives

$$\dot{m}_{ij} = -m_{ij} + S_i S_j + [x_i y_j - x_i S_j - y_j S_i] \quad (10)$$

which is equivalent to the signal Hebb law (1) if and only if the term in braces is zero. Thus the simple differential Hebb law (9), and of course (4) suitably scaled, reduces to the signal Hebb law when no pulses occur, when $x_i(t) = y_j(t) = 0$. This happens frequently. For, in any connected time interval, the set ν of times where pulses occur, $\{t': x_i(t') = 1\}$, has Lebesgue measure zero. (Consider pulses at rational time points or at Cantor set points.) This interpretation, though, would imply [38] by (5) and (6) that $S_i = S_j = 0$ almost everywhere, so the integrals in (5) and (6) would have to be replaced with discrete sums (using point-mass measures).

The infrequency of unit pulses occurs while the synapse m_{ij} continually modifies its behavior. When instantaneous pulse information is not available, the synapse "fills in" with expected pulse frequencies, and hence Hebbian learning. Since signal Hebbian learning is unconditionally stable (the ABAM theorem, reviewed below) in many nonlinear dynamical systems, including popular feedback neural networks, pulse-coded differential Hebbian dynamical systems may be stable over a wider range of system parameters than earlier velocity-acceleration stability assumptions [32], [33] suggested.

D. Differential Competitive Learning

The fourth unsupervised learning law is a new hybrid learning law, the *differential competitive* law:

$$\dot{m}_{ij} = \dot{S}_j [S_i - m_{ij}]. \quad (11)$$

The idea is *learn only if change*. As with the competitive learning law (3), the neurons in F_Y compete for acti-

vation, and the nonnegative signal functions S_j keep score. The signal velocity \dot{S}_j in (11) is a local *reinforcement* mechanism. Its sign indicates whether the j th neurons are winning or losing, and its magnitude measures by how much. The coding and dynamical behavior of (11) can be analyzed with the pulse-coding interpretation [10], [11] of signal functions and by comparison with Kohonen's recent "supervised" adaptive-vector-quantization algorithm [24].

The pulse-coded differential competitive learning law is the difference of nondifferential competitive laws:

$$\dot{m}_{ij} = (y_j - S_j)(S_i - m_{ij}) \quad (12)$$

$$= y_j[S_i - m_{ij}] - S_j[S_i - m_{ij}] \quad (13)$$

where x_i is a 0-1 pulse function. Hence the standard competitive learning law (3) is recovered when $y_j = 1$ and $S_j = 0$. This occurs when the j th unit has just won the competition for activation within F_Y .

Usually in a competition there are many more losers than winners. So suppose the j th neuron in F_Y is a loser at time t . Then $y_j(t) = 0$ holds and has held over some, perhaps short, past interval $[t', t]$. Then $S_j(t) = 0$ (or nearly 0) by the exponential-weight structure of (6). So no change, no learning.

Now suppose the j th unit wins in the next instant t . Then $y_j = 1$ over some interval $[t, t']$ of nonzero Lebesgue measure. During this interval the exponential-weight structure of S_j soon drives S_j toward 1, which we take as the upper bound of S_j . This means m_{ij} quickly approaches a positively scaled version of the signal S_i .

Now suppose the j th unit goes from winning to losing. Then at first $y_j = 0$ and $S_j = 1$. As S_j quickly falls to zero, learning slows then stops when $y_j = S_j = 0$. Meanwhile m_{ij} has "moved away" from the signal S_i . The signal velocity \dot{S}_j has "punished" the j th unit.

Kohonen [24] uses a sign change to punish misclassifying prototype vectors trained with the competitive learning law in his feedforward "supervised" adaptive vector quantization (AVQ) system. In vector formulation, the p reference vectors $m_1(t), \dots, m_p(t)$ are the respective prototypes at time t of the p decision classes D_1, \dots, D_p that partition the signal space R^n . The p reference vectors are also the p columns of the synaptic matrix M . $m_i = (m_{i1}, \dots, m_{in})$ is the fan-in of synapses of the i th neuron in F_Y . All F_Y neurons are engaged in winner-take-all competition. Given a random training sample vector $x(t)$ presented at F_X , the F_Y competition is summarized by finding the reference vector $m_j(t)$ closest to $x(t)$ in Euclidean distance: $\|x - m_j\| = \min \{\|x - m_i\| : i = 1, \dots, p\}$. "Supervision" means we know which decision class the random vector x was chosen from. If x belongs to D_j , the class represented by m_j , then m_j is rewarded by moving m_j a little closer to x . This allows m_j to gradually approximate the centroid of D_j . (The centroid, or conditional expectation, minimizes the mean-squared-error of vector quantization [37].) Else if x does not belong to D_j , m_j is punished for misclassifying x as a

D_j pattern by moving m_j a little farther away from x , presumably out of regions of misclassification. This is achieved by a simple sign change:

$$m_j(t+1) = \begin{cases} m_j(t) + c(t)[x(t) - m_j(t)], & x \in D_j \\ m_j(t) - c(t)[x(t) - m_j(t)], & x \notin D_j \end{cases} \quad (14)$$

$$m_i(t+1) = m_i(t) \text{ for all losing neurons in } F_Y \quad (15)$$

where $c(0), c(1), c(2), \dots$ is a slowly decreasing sequence of small ($c(0) < 1$) learning constants. Kohonen's "unsupervised" AVQ algorithm eliminates the punishment equation (15) and relaxes (14) by allowing $x(t)$ to belong to any decision class. The unsupervised algorithm is clearly a discrete stochastic version of the competitive law (3) in vector notation. Kohonen shows that under appropriate statistical conditions, the equilibrium condition of the AVQ unsupervised-clustering algorithm occurs when the p reference vectors m_i asymptotically arrive at the centroids of their respective decision classes. Kohonen next shows that the equilibrium condition of the supervised AVQ algorithm is similar in structure to that of the optimum unit-cost Bayes classifier, and cites simulation data in support of this similarity.

The differential competitive law (11) can be viewed as a local unsupervised *approximation* of Kohonen's supervised AVQ algorithm. Indeed preliminary simulations of (11) in stochastic feedforward mode show similar classification performance in many noise environments.

The pulse-coded differential competitive law (12), as discussed above, can be expected to often behave as the competitive law (3) with 0-1 threshold signal function S_j . This is precisely when the competitive law has been shown [32] globally stable when embedded in the nonlinear dynamical systems below. For this reason, we here limit the stability analysis of the differential competitive law to that of the competitive law with steep signal function S_j . We similarly limit the stability analysis of the differential Hebb law (4) to the analysis of the signal Hebb law, even though differential Hebb dynamical systems are known [32], [33] globally stable in the special case that signal velocities are comparable to signal accelerations.

III. UNIDIRECTIONAL AND BIDIRECTIONAL NONLINEAR DYNAMICAL SYSTEMS

We study nonlinear dynamical systems described by Cohen-Grossberg [6], [14] dynamics. In the unidirectional or autoassociative case, when $F_X = F_Y$ and $M = M^T$, a neural network possesses Cohen-Grossberg dynamics if its activation equations can be written in the abstract form

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j(x_j) m_{ij} \right] \quad (17)$$

where $a_i(x_i) \geq 0$ is an amplification function, b_i is arbitrary so long as it keeps the integrals bounded in the Lyapunov functions below, and S_i is a bounded monotone nondecreasing ($S'_i \geq 0$) signal function. The global stability of nonlearning autoassociative systems described by (17) is ensured by the Cohen-Grossberg theorem [6], which is abstractly equivalent—in the sense that $R^n \times R^p = R^{n+p}$ —to the BAM theorem below for nonlearning heteroassociative networks and a special case of the ABAM theorem reviewed in the next section.

Perhaps the most important special cases of (17) are additive and shunting networks, the popular versions of which are the respective Hopfield circuit [19] and the Hodgkin-Huxley membrane equation [18]. Grossberg [14], has also shown that (17) reduces to the additive brain-state-in-a-box model of Anderson [1], [2] and the shunting masking field model [7] upon appropriate change of variables. An autoassociative system has *additive* activation dynamics if the amplification function a_i is constant and the b_i function is linear. For instance, if $a_i = 1/C_i$, $b_i = (x_i/R_i) - I_i$, $S_i(x_i) = g_i(x_i) = V_i$, and constant $m_{ij} = m_{ji} = T_{ij} = T_{ji}$, where C_i and R_i are positive constants and input I_i is constant or slowly varying relative to fluctuations in x_i , then (17) reduces to the Hopfield circuit [19]:

$$C_i \dot{x}_i = -\frac{x_i}{R_i} + \sum_j V_j T_{ij} + I_i. \quad (18)$$

Grossberg [13] has shown that neurons with additive dynamics saturate at their upper bounds (if they have them) when inputs are arbitrarily large, thus ignoring the relative pattern information in the input pattern (I_1, \dots, I_n).

An autoassociative network has *shunting* or multiplicative activation dynamics when the amplification function a_i is linear and b_i is nonlinear. For instance, if $a_i = -x_i$, $m_{ij} = 1$ (self-excitation in lateral inhibition), and $b_i = (1/x_i)[-A_i x_i + B_i(S_i + I_i^+) - x_i(S_i + I_i^+) - C_i(\sum_{j \neq i} S_j m_{ij} + I_i^-)]$, gives the distance-dependent ($m_{ij} = m_{ji}$) unidirectional shunting network:

$$\begin{aligned} \dot{x}_i = & -A_i x_i + (B_i - x_i)[S_i(x_i) + I_i^+] \\ & - (C_i + x_i) \left[\sum_{j \neq i} S_j(x_j) m_{ij} + I_i^- \right] \end{aligned} \quad (19)$$

where A_i is a positive decay constant and B_i and C_i are positive saturation constants. The first term on the right-hand side of (19) is a passive decay term. The second and third terms are, respectively, positive and negative feedback terms. (Strictly speaking, $a_i(x_i)$ must be kept positive. x_i can always be translated to achieve this.) If the shunting x_i terms in the positive and negative feedback terms are scaled to zero, (19) reduces to an additive model. Grossberg also showed that shunting models do not saturate when presented with arbitrarily large positive inputs. They remain sensitive to the relative pattern information in (I_1, \dots, I_n). Perhaps more important for

neurobiologists, Grossberg [13], [14] observed that the shunting model (19) is naturally generalized by the celebrated Hodgkin-Huxley membrane equation:

$$c \frac{\partial V_i}{\partial t} = (V^p - V_i)g_i^p + (V^+ - V_i)g_i^+ + (V^- - V_i)g_i^- \quad (20)$$

where V^p , V^+ , and V^- are respective passive, excitatory (sodium Na^+), and inhibitory (potassium K^+) saturation upper bounds with corresponding shunting conductances g_i^p , g_i^+ , and g_i^- , and where the constant capacitance $c > 0$ scales time. The shunting model (19) becomes the membrane equation (20) if $V_i = x_i$, $V^p = 0$, $V^+ = B_i$, $V^- = -C_i$, $g_i^p = A_i$, $g_i^+ = S_i(x_i) + I_i^+$, and $g_i^- = \sum_{j \neq i} S_j m_{ij} + I_i^-$.

Continuous *bidirectional associative memories* [28]–[32] (BAM's) arise when two (or more) neural fields F_X and F_Y are connected in the forward direction, from F_X to F_Y , by an arbitrary n -by- p synaptic matrix M and connected in the backward direction, from F_Y to F_X , by the p -by- n matrix $N = M^T$, where M^T is the transpose of M . BAM activations also possess Cohen-Grossberg dynamics, and their extensions:

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_{j=1}^p S_j(y_j) m_{ij} \right] \quad (21)$$

$$\dot{y}_j = -a_j(y_j) \left[b_j(y_j) - \sum_{i=1}^n S_i(x_i) m_{ij} \right] \quad (22)$$

with corresponding Lyapunov function L :

$$\begin{aligned} L = & -\sum_i \sum_j S_i S_j m_{ij} + \sum_i \int_0^{x_i} S'_i(\theta_i) b_i(\theta_i) d\theta_i \\ & + \sum_j \int_0^{y_j} S'_j(\epsilon_j) b_j(\epsilon_j) d\epsilon_j \end{aligned}$$

where the functions b_i and b_j must be suitably constrained to keep L bounded.

The quadratic form in L is bounded because the signal functions S_i and S_j are bounded. Boundedness of the integral terms requires additional technical hypotheses to avoid pathologies as discussed by Cohen and Grossberg [6]. For our purpose we simply assume the integral terms are bounded.

All BAM results extend to any number of BAM-connected fields. Complex topologies are possible and, in theory, will equilibrate as rapidly as the two-layer BAM system. The back-and forth flow of information in a BAM facilitates natural large-scale optical implementations [20], [28].

The BAM model (21), (22) clearly reduces to the Cohen-Grossberg model if both neural fields collapse into one, $F_X = F_Y$, and the constant matrix M is symmetric ($M = M^T$). Conversely, the BAM system, which is always globally stable, can be abstractly viewed [30] as symmetrizing an arbitrary matrix M . For if the two BAM fields

are abstractly concatenated into a new field F_Z , $F_Z = F_X \cup F_Y$, with zero block diagonal synaptic matrix W that contains M and M^T as respective upper and lower blocks, then the BAM dynamical system (21), (22) is equivalent to the autoassociative system (17).

The BAM system (21) includes additive and shunting models. If $a_i = 1 = a_j$, $b_i = x_i - I_i$, and $b_j = y_j - J_j$, for relatively constant inputs I_i and J_j , then an *additive* BAM [30], [31] results:

$$\dot{x}_i = -x_i + \sum_j S_j(y_j) m_{ij} + I_i \quad (23)$$

$$\dot{y}_j = -y_j + \sum_i S_i(x_i) m_{ij} + J_j \quad (24)$$

where again constants can be added or multiplied as desired. More generally, if $a_i = -x_i$, $a_j = -y_j$, $b_i = (1/x_i)[-x_i + (B_i - x_i)[S_i(x_i) + I_i^+] - x_i I_i^-]$, and $b_j = (1/y_j)[-y_j + (B_j - y_j)[S_j(y_j) + J_j^+] - y_j J_j^-]$, then a *shunting* BAM [30] results:

$$\dot{x}_i = -x_i + (B_i - x_i)[S_i + I_i^+] - x_i \left[\sum_j S_j m_{ij} + I_i^- \right] \quad (25)$$

$$\dot{y}_j = -y_j + (B_j - y_j)[S_j + J_j^+] - y_j \left[\sum_i S_i m_{ij} + J_j^- \right]. \quad (26)$$

The shunting BAM (25), (26) reminds us that in general distance-dependent competition occurs within fields F_X and F_Y . Suppose the n -by- n matrix R and the p -by- p matrix S describe the distance-dependent ($R = R^T$, $S = S^T$) lateral inhibition within F_X and F_Y , respectively. Then the general BAM model (21), (22) must be augmented to a competitive BAM [29]:

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j^p S_j(y_j) m_{ij} - \sum_k^n S_k(x_k) r_{ki} \right] \quad (27)$$

$$\dot{y}_j = -a_j(y_j) \left[b_j(y_j) - \sum_i^n S_i(x_i) m_{ij} - \sum_l^p S_l(y_l) s_{lj} \right]. \quad (28)$$

An *adaptive* bidirectional associative memory (ABAM) is a globally stable dynamical system with activation dynamics described by (21), (22) or (27), (28) and synaptic dynamics described by a first-order learning law. The original ABAM [30] restricted the choice of learning law to the signal Hebb law (1). Signal Hebb ABAM's are unconditionally globally stable, though limited in their ability to estimate continuous functions. Better, though more costly, estimation can be gotten with higher order signal Hebb ABAM's. For example, in autoassociative notation, the second-order signal Hebb ABAM [32] is described by

(29)–(31):

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j(x_j) m_{ij} - \sum_j \sum_k S_j(x_j) S_k(x_k) n_{ijk} \right] \quad (29)$$

$$\dot{m}_{ij} = -m_{ij} + S_i(x_i) S_j(x_j) \quad (30)$$

$$\dot{n}_{ijk} = -n_{ijk} + S_i(x_i) S_j(x_j) S_k(x_k) \quad (31)$$

with corresponding Lyapunov function L :

$$\begin{aligned} L = & -\frac{1}{2} \sum_i \sum_j S_i S_j m_{ij} - \frac{1}{3} \sum_i \sum_j \sum_k S_i S_j S_k n_{ijk} \\ & + \sum_i \int_0^{x_i} S'_i(\theta_i) b_i(\theta_i) d\theta_i + \frac{1}{4} \sum_i \sum_j m_{ij}^2 \\ & + \frac{1}{6} \sum_i \sum_j \sum_k n_{ijk}^2. \end{aligned} \quad (32)$$

The Lyapunov function remains bounded in the adaptive case. The new terms

$$\frac{1}{4} \sum_i \sum_j m_{ij}^2 \quad \text{and} \quad \frac{1}{6} \sum_i \sum_j \sum_k n_{ijk}^2 \quad (33)$$

in (32) are bounded because the solutions to (30) and (31) are bounded since, ultimately, the signal functions S_i are bounded.

If $a_i(x_i) > 0$ and $S'_i > 0$, and if (32) is differentiated with respect to time, rearranged, and (29), (30) are used to eliminate terms, then L strictly decreases along trajectories, yielding asymptotic stability (and in general exponential convergence), since

$$\begin{aligned} \dot{L} = & -\sum_i \frac{S'_i(x_i)}{a_i(x_i)} \dot{x}_i^2 - \frac{1}{2} \sum_i \sum_j \dot{m}_{ij}^2 \\ & - \frac{1}{3} \sum_i \sum_j \sum_k \dot{n}_{ijk}^2 < 0 \end{aligned} \quad (34)$$

if any activation or synaptic velocity is nonzero. The strict monotonicity assumption $S'_i > 0$ and (33) further imply that $\dot{L} = 0$ if and only if all parameters stop changing: $\dot{x}_i = \dot{m}_{ij} = \dot{n}_{ijk} = 0$ for all i, j, k . All like higher order ABAM's are globally stable.

The restriction to signal Hebbian learning was relaxed [32] to allow competitive learning with (3) provided S_j is steep, and further relaxed to allow differential Hebbian learning with (4) provided signal velocities and signal accelerations agree in sign. A *competitive* ABAM (CABAM) results from (27), (28) if learning is governed by the competitive learning law (3) and if S_j behaves essentially as a 0-1 step function. For then, upon time differentiation, the appropriate Lyapunov function L takes the form

$$\begin{aligned} \dot{L} = & -\sum_i \frac{S'_i(x_i)}{a_i} \dot{x}_i^2 - \sum_j \frac{S'_j(y_j)}{a_j} \dot{y}_j^2 \\ & - \sum_i \sum_j \dot{m}_{ij} [S_i(x_i) S_j(y_j) - m_{ij}]. \end{aligned} \quad (35)$$

The trick is to eliminate \dot{m}_{ij} in (34) with the competitive law (3) and exploit the 0-1 threshold (steep-sigmoid) behavior of S_j . Then the relevant product becomes non-negative:

$$\begin{aligned} \dot{m}_{ij}[S_i S_j - m_{ij}] &= S_j(S_i - m_{ij})[S_i S_j - m_{ij}] \\ &= \begin{cases} 0, & S_j(y_j) = 0 \\ (S_i - m_{ij})^2, & S_j(y_j) = 1. \end{cases} \end{aligned}$$

Thus both winners and losers in F_Y keep L decreasing and ensure that every CABAM is globally stable.

CABAM's are topologically equivalent to *adaptive resonance theory* (ART) systems [13]. The idea behind ART systems is *learn only if resonate*. Resonance, though, is simply joint stability at F_X and F_Y mediated by the forward connections M and the backward connections N . When $N = M^T$ and activation dynamics are described by (27), (28), ART models become CABAM models so long as learning is described by a globally stable learning law, in particular the competitive law (3) with steep signal function S_j . This is the case with the recent ART-2 model [5] since the activation (short-term memory) dynamics of F_X and F_Y are described by shunting equations and, in the notation of Carpenter and Grossberg, the learning (long-term memory) dynamics are described by CABAM-style competitive learning laws with threshold signal functions in F_Y :

$$\text{top-down } (F_Y \rightarrow F_X): \dot{z}_{ji} = g(y_j)[p_i - z_{ji}] \quad (36)$$

$$\text{bottom-up } (F_X \rightarrow F_Y): \dot{z}_{ij} = g(y_j)[p_i - z_{ij}] \quad (37)$$

where g is a threshold signal function and p_i is the signal pattern (itself involving complicated L^2 -norm computations) transmitted from F_X . Equation (36) says matrix Z contains forward projections and its transpose Z^T contains backward connections.

In contrast, the earlier binary ART-1 model [4] is not *extended* by the CABAM model because Weber law structure is imposed on the forward "bottom-up" synaptic projections, and thus the forward and backward connection matrices are not related by transposition. This in part explains why binary inputs in ART-2 need not produce ART-1 behavior. It also suggests that the ART-2 model can in principle be similarly modified by adding Weber law structure to (36), producing an ART-2' model that is not a CABAM.

These connections among unsupervised feedback dynamical systems are summarized by the taxonomy in Fig. 2 of artificial neural networks (ANN's) and placed in context with unsupervised feedforward adaptive vector quantizers and the extremely popular supervised feedforward gradient-descent networks:

The more general RABAM model is developed below.

Finally, for completeness, we state the form of ABAM systems that adapt (and activate) with signal velocity in-

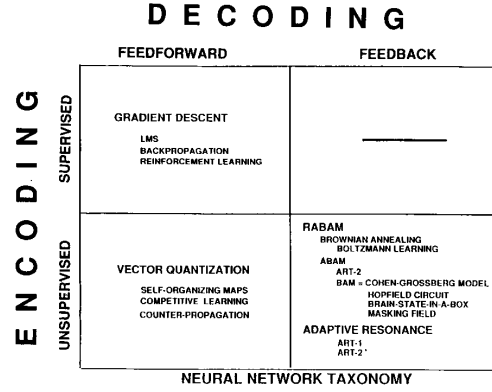


Fig. 2.

formation by using the differential Hebb learning law [33]:

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j m_{ij} - \sum_j \dot{S}_j m_{ij} \right] \quad (38)$$

$$\dot{y}_j = -a_j(y_j) \left[b_j(y_j) - \sum_i S_i m_{ij} - \sum_i \dot{S}_i m_{ij} \right] \quad (39)$$

$$\dot{m}_{ij} = -m_{ij} + S_i S_j + \dot{S}_i \dot{S}_j \quad (40)$$

and the further assumptions $\dot{S}_i \approx \ddot{S}_i$, $\dot{S}_j \approx \ddot{S}_j$, where in general (40) can be loosened to only require that signal velocities and accelerations tend to have the same sign (as in clipped exponentials). The corresponding Lyapunov function now includes a "kinetic energy" term to account for signal velocities:

$$\begin{aligned} L &= - \sum_i \sum_j S_i S_j m_{ij} - \sum_i \sum_j \dot{S}_i \dot{S}_j m_{ij} \\ &\quad + \sum_i \int_0^{x_i} S'_i(\theta_i) b_i(\theta_i) d\theta_i \\ &\quad + \sum_j \int_0^{y_j} S'_j(\epsilon_j) b_j(\epsilon_j) d\epsilon_j + \frac{1}{2} \sum_i \sum_j \dot{m}_{ij}^2. \end{aligned}$$

IV. STABILITY-CONVERGENCE DILEMMA AND THE ABAM THEOREM

Stability and convergence are equilibrium properties. *Stability* is equilibrium in a neuronal field: $(d/dt)F_X = 0$. *Convergence* is equilibrium in a synaptic web: $(d/dt)M = 0$. *Global stability* is joint stability and convergence for all inputs and all network parameters. *Pattern formation* occurs across field F_X when it stabilizes. The stable signals across F_X make up the formed pattern. Stability is trivial in a feedforward network.

Global stability is difficult to achieve in unsupervised feedback networks. After all, most feedback systems are unstable. Global stability requires a delicate dynamical balance between stability and convergence. Achieving such a balance is arguably the central problem in analyzing, and building, unsupervised feedback dynamical sys-

tems. The chief difficulty stems from the dynamical asymmetry between neural and synaptic fluctuations. Neurons fluctuate orders of magnitude faster than synapses: learning is slow. In real neural systems, neuronal fluctuation may be at the millisecond level, while synaptic fluctuation may be at the second or even minute level.

The *stability-convergence dilemma* arises from the asymmetry in neuronal and synaptic fluctuation rates. The dilemma unfolds as follows. Neurons change faster than synapses change. Patterns form when neurons stabilize, when $(d/dt)F_X = 0$ and $(d/dt)F_Y = 0$. The slowly varying synapses M try to learn these patterns. Since the neurons are stable for more than a synaptic moment, the synapses begin to adapt to the neuronal patterns—learning begins. So $(d/dt)F_X = 0$ and $(d/dt)F_Y = 0$ imply $(d/dt)M \neq 0$. Since there are numerous feedback paths from the synapses to the neurons, the neurons tend to change state. So $(d/dt)M \neq 0$ implies $(d/dt)F_X \neq 0$ and $(d/dt)F_Y \neq 0$. *Learning tends to undo the very stability patterns to be encoded*, and hence the dilemma. In summary, for two fields of neurons F_X and F_Y connected in the forward direction by M and in the backward direction by M^T , the stability-convergence dilemma has four parts, described as follows.

A. Stability-Convergence Dilemma

1) *Asymmetry*: Neurons in F_X and F_Y fluctuate faster than the synapses M .

2) *Stability*: $\frac{d}{dt}F_X = 0$ and $\frac{d}{dt}F_Y = 0$ (pattern formation).

3) *Learning*: $\frac{d}{dt}F_X = 0$ and $\frac{d}{dt}F_Y = 0 \rightarrow \frac{d}{dt}M \neq 0$.

4) *Undoing*: $\frac{d}{dt}M \neq 0 \rightarrow \frac{d}{dt}F_X \neq 0$ and $\frac{d}{dt}F_Y \neq 0$.

The ABAM theorem [32] provides one resolution of the stability-convergence dilemma. The adaptive resonance concept provides another. Though as discussed in the previous section, the recent ART-2 instantiation of the concept is a CABAM. The ABAM theorem ensures the global stability, the joint stability and convergence, of dynamical systems with activation dynamics described by (21) and (22) and that learn according to the signal Hebb learning law (1). The extensions to competitive and differential Hebbian learning (and thus differential competitive learning) discussed above all require more assumptions than learning with the signal Hebb law, which requires none. Since the ABAM theorem is the starting point for the random-process extension to the RABAM theorem below, we review its statement and proof.

B. ABAM Theorem

Every signal Hebb BAM is asymptotically stable, where the network dynamics are described by

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j(y_j) m_{ij} \right] \quad (41)$$

$$\dot{y}_j = -a_j(y_j) \left[b_j(y_j) - \sum_i S_i(x_i) m_{ij} \right] \quad (42)$$

$$\dot{m}_{ij} = -m_{ij} + S_i(x_i) S_j(y_j) \quad (43)$$

and $a_i > 0$ and $a_j > 0$, and S_i and S_j are bounded monotone increasing ($S'_i > 0$ and $S'_j > 0$) signal functions. At equilibrium, all activation and synaptic velocities are zero.

Proof. Consider the global Lyapunov function L :

$$L = -\sum_i \sum_j S_i S_j m_{ij} + \sum_i \int_0^{x_i} S'_i(\theta_i) b_i(\theta_i) d\theta_i + \sum_j \int_0^{y_j} S'_j(\epsilon_j) b_j(\epsilon_j) d\epsilon_j + \frac{1}{2} \sum_i \sum_j m_{ij}^2. \quad (44)$$

Then time differentiation and collection of like terms gives

$$\dot{L} = -\sum_i S'_i \dot{x}_i \left[b_i - \sum_j S_j m_{ij} \right] + \sum_j S'_j \dot{y}_j \left[b_j - \sum_i S_i m_{ij} \right] - \sum_i \sum_j \dot{m}_{ij} [S_i S_j - m_{ij}]. \quad (45)$$

Then, using the positivity of a_i and a_j , the terms in braces can be eliminated with the respective equations (41)–(43). This proves that L is strictly decreasing along trajectories:

$$\dot{L} = -\sum_i \frac{S'_i}{a_i} \dot{x}_i^2 - \sum_j \frac{S'_j}{a_j} \dot{y}_j^2 - \sum_i \sum_j \dot{m}_{ij}^2 < 0 \quad (46)$$

for any activation or synaptic change. Since $S'_i > 0$ and $S'_j > 0$, $\dot{L} = 0$ if and only if $\dot{x}_i = \dot{y}_j = \dot{m}_{ij} = 0$ for all i and j . Q.E.D.

The strictly inequality sign in (46) yields asymptotic stability, which ensures that trajectories end in equilibrium points, not merely near them. Asymptotic stability also ensures that the eigenvalues of the Jacobian matrix of the system (41)–(43) have nonpositive real parts near equilibria. A nondegenerate Hessian further ensures that the real parts of the eigenvalues are negative. Then [16] the nonlinear system (41)–(43) converges exponentially quickly as if it were linear.

V. RANDOM ADAPTIVE BIDIRECTIONAL ASSOCIATIVE MEMORIES

Random adaptive bidirectional associative memory (RABAM) models are everywhere perturbed by Brownian diffusions. The differential equations in (41)–(43) now become stochastic differential equations, with random processes as solutions. In the simplest case, Brownian diffusions are simply added to deterministic differential equations. In the more general case adopted here, every activation and synaptic variable represents a separate stochastic process. The stochastic differential equations relate the time evolution of these stochastic processes. Brownian diffusions, or “noise” processes, are then added to the stochastic differential equations. In principle this Ito calculus approach need not preserve the chain rule of deterministic differential calculus. The final section,

though, discusses why for RABAM models the classical chain-rule relationships still hold.

Let B_i , B_j , and B_{ij} be Brownian motion (independent Gaussian increment) processes [35], [41] perturbing the i th neuron in F_X , the j th neuron in F_Y , and the synapse m_{ij} , respectively. The Brownian motions are allowed to have time-varying diffusion parameters. Then the *diffusion* RABAM is described by (47)–(49):

$$dx_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j(y_j) m_{ij} \right] dt + dB_i \quad (47)$$

$$dy_j = -a_j(y_j) \left[b_j(y_j) - \sum_i S_i(x_i) m_{ij} \right] dt + dB_j \quad (48)$$

$$dm_{ij} = -m_{ij} dt + S_i(x_i) S_j(y_j) dt + dB_{ij}. \quad (49)$$

The signal Hebb diffusion law (49) can be replaced with the competitive diffusion law

$$dm_{ij} = S_j(y_j) [S_i - m_{ij}] dt + dB_{ij} \quad (50)$$

if S_j is sufficiently steep. Or it can be replaced with differential Hebb or differential competitive diffusion laws if tighter constraints are imposed. For simplicity, we shall formulate the RABAM model in the signal Hebb case only. The extensions to competitive and differential learning proceed exactly as the above extensions of the ABAM theorem. All RABAM results, like all ABAM results, also immediately extended to high-order systems of arbitrarily high order.

The RABAM model can be restated in more familiar, less rigorous, “noise notation.” Intuitively independent zero-mean noise is added to the ABAM model. The stochastic differential equations then describe the time evolution of network “signals plus noise.” This implicitly means that the noise processes are independent of the nonlinear “signal” processes. For emphasis, though, we explicitly make the weaker assumption that the noise processes are *uncorrelated* with the “signal” processes. We further assume that the noise processes have finite variances, though they may be time varying. Then the *noise* RABAM model is described by the stochastic differential equations

$$\dot{x}_i = -a_i(x_i) \left[b_i(x_i) - \sum_j S_j(y_j) m_{ij} \right] + n_i \quad (51)$$

$$\dot{y}_j = -a_j(y_j) \left[b_j(y_j) - \sum_i S_i(x_i) m_{ij} \right] + n_j \quad (52)$$

$$\dot{m}_{ij} = -m_{ij} + S_i(x_i) S_j(y_j) + n_{ij} \quad (53)$$

$$E(n_i) = E(n_j) = E(n_{ij}) = 0 \quad (54)$$

$$V(n_i) = \sigma^2 n_i < \infty, \quad \sigma_j^2 < \infty, \quad \sigma_{ij}^2 < \infty. \quad (55)$$

Noise can be added within the general b_i and b_j terms, perhaps reflecting random input signals. A separate analysis [34] shows that additive input noise can be accommodated for additive and shunting activation models. For

additive activation models, such additive activation noise can be included in the noise terms n_i and n_j .

Will so much noise destabilize the system? So much noise with so much feedback would seem to promote chaos, especially since the network dimensions n and p can be arbitrarily large. How can stable learning occur?

The RABAM theorem ensures stochastic stability. Nonlinear interactions suppress noise and suppress it exponentially quickly. In effect, RABAM equilibria are ABAM equilibria that randomly vibrate. The diffusion parameters, or the noise variances, control the range of vibration. Average RABAM behavior is just ABAM behavior. Since noise perturbations do not destroy equilibria, the RABAM theorem says that unsupervised learning is structurally stable in the stochastic sense. The result applies with equal force, though with less theoretical interest, for unsupervised learning in feedforward networks.

The RABAM theorem can be motivated with a simple thought experiment or, better, a few hand calculations. Consider a discrete additive BAM with fixed matrix M . Find its bipolar fixed points in the product space $\{-1, 1\}^n \times \{-1, 1\}^p$. Now add a small amount of zero-mean noise to each memory element m_{ij} . Since a discrete BAM signal function is a threshold function, it is unlikely that more than very few neurons, if any, change state differently during iterations than they did before. It is even less likely that they will do so as n and p increase. The same fixed points tend to be reached, and tend to persist once reached. This corresponds to adding noise at the synaptic level. Now repeat the computation, but also add zero-mean noise to each neuron's activation at each iteration. Then repeat this computation, adding new noise to the matrix M each time. This allows the synaptic noise processes to be “lower” than the neuronal noise processes. Again the threshold signal functions make it unlikely that the signal patterns will change significantly, if at all, during iterations or in equilibrium.

A. RABAM Theorem

The RABAM model (47)–(50), or (51)–(55) is globally stable. If signal functions are strictly increasing and amplification functions a_i and a_j are strictly positive, the RABAM model is asymptotically stable.

Proof. The ABAM Lyapunov function (44) is now a random process. At each time t , $L(t)$ is a random variable. We conjecture that *the expected ABAM Lyapunov function $E(L)$ is a Lyapunov function for the RABAM system*, where the expectation is with respect to all random parameters:

$$E(L) = \int \cdots \int L p(X, Y, M) dX dY dM. \quad (56)$$

(Recall that each activation and synaptic parameter represents a random process separate from the random process got simply by adding noise to a deterministic variable.)

The proof strategy is to replace the time derivative of the expectation with the expectation of the time derivative

of the ABAM Lyapunov function, which we calculated above. Technically we need to assume sufficient smoothness conditions on the RABAM model to bring the time derivative inside the multiple integrals in (56). This assumption adds little burden. Then

$$\begin{aligned}
 \dot{E}(L) &= E(\dot{L}) \text{ and by (45)} \\
 &= E \left\{ \sum_i S'_i \dot{x}_i \left[b_i - \sum_j S_j m_{ij} \right] \right. \\
 &\quad + \sum_j S'_j \dot{y}_j \left[b_j - \sum_i S_i m_{ij} \right] \\
 &\quad \left. - \sum_i \sum_j \dot{m}_{ij} [-m_{ij} + S_i S_j] \right\} \\
 &= E \left\{ - \sum_i S'_i a_i \left[b_i - \sum_j S_j m_{ij} \right]^2 \right. \\
 &\quad - \sum_j S'_j a_j \left[b_j - \sum_i S_i m_{ij} \right]^2 \\
 &\quad \left. - \sum_i \sum_j [-m_{ij} + S_i S_j]^2 \right\} \\
 &\quad + \sum_i E \left\{ n_i S'_i \left[b_i - \sum_j S_j m_{ij} \right] \right\} \\
 &\quad + \sum_j E \left\{ n_j S'_j \left[b_j - \sum_i S_i m_{ij} \right] \right\} \\
 &\quad - \sum_i \sum_j E \left\{ n_{ij} [-m_{ij} + S_i S_j] \right\} \quad (57)
 \end{aligned}$$

upon eliminating the activation and synaptic velocities in (57) with the RABAM dynamical equations (51)–(53)

$$\begin{aligned}
 &= E[\dot{L}_{\text{ABAM}}] + \sum_i E(n_i) E \left\{ S'_i \left[b_i - \sum_j S_j m_{ij} \right] \right\} \\
 &\quad + \sum_j E(n_j) E \left\{ S'_j \left[b_j - \sum_i S_i m_{ij} \right] \right\} \\
 &\quad - \sum_i \sum_j E(n_{ij}) E[-m_{ij} + S_i S_j] \quad (58)
 \end{aligned}$$

by the uncorrelatedness (independence) of the “signal” and additive noise terms in the RABAM model, and by the facts that S'_i and S'_j are nonnegative functions of x_i and y_j respectively, and a_i and a_j are nonnegative essentially arbitrary functions (so $S'_i = a_i$ and $S'_j = a_j$ possible)

$$= E[\dot{L}_{\text{ABAM}}]$$

by (54). So $\dot{E}(L) \leq 0$ or $\dot{E}(L) < 0$ along trajectories according as $\dot{L}_{\text{ABAM}} \leq 0$ or $\dot{L}_{\text{ABAM}} < 0$. Q.E.D.

VI. NOISE-SATURATION DILEMMA AND THE RABAM NOISE SUPPRESSION THEOREM

How much do RABAM trajectories and equilibria vibrate? To answer this question we need to examine the *second-order* behavior of the RABAM model. This be-

havior depends fundamentally on the variances of the additive noise processes. Observe that the zero-mean assumption (54) implies that the time-varying “variances” σ_i^2 , σ_j^2 , and σ_{ij}^2 are the respective instantaneous mean-squared “noises” $E(n_i^2)$, $E(n_j^2)$, and $E(n_{ij}^2)$, since in general $V(x) = E(x^2) - E^2(x)$.

Observed RABAM second-order behavior consists of the observed instantaneous *mean-squared velocities* $E(\dot{x}_i^2)$, $E(\dot{y}_j^2)$, and $E(\dot{m}_{ij}^2)$. The mean-squared velocities measure the magnitude of instantaneous RABAM change. They are at least as large as the underlying instantaneous “variances” of the activation velocity and synaptic velocity processes, since, for example

$$E(\dot{x}_i^2) \geq E(\dot{x}_i^2) - E^2(\dot{x}_i) = V(\dot{x}_i). \quad (59)$$

Intuitively the mean-squared velocities should depend on the instantaneous “variances” of the noise processes in (51)–(53). The more the noise processes hop about their means, the greater the potential for the activations and synapses to change state. But this intuition seems to run counter to the structural stability established by the RABAM theorem. Surely, it seems, if the magnitudes of the noise fluctuations grow arbitrarily large, there comes a point—and perhaps a point quickly reached in the midst of massive noisy feedback—where the RABAM system transitions from stability to instability.

The RABAM noise suppression theorem guarantees that no noise processes can destabilize a RABAM if the noise processes have *finite* instantaneous variances. (Cauchy noise, for example, in theory could destabilize a RABAM since it has infinite variance. In practice, though, even Cauchy variance is finite, and so it will never destabilize a RABAM.) Preliminary simulations [43], where noise fluctuations are many orders of magnitude greater than activation and synaptic fluctuations, have confirmed this surprising prediction. In some sense noise cannot beat RABAM stability. Moreover, the RABAM noise suppression theorem ensures that noise will be “quenched,” to use Grossberg’s term [13], exponentially quickly in most cases.

To prove the RABAM noise suppression theorem, we must make explicit how RABAM instantaneous mean-squared velocities depend on the underlying instantaneous noise variances. The following lemma grounds the intuition that observed second-order behavior—the instantaneous mean-squared velocities—involves at least as much fluctuation as is found in the noise itself.

Lemma:

$$E(\dot{x}_i^2) \geq \sigma_i^2, \quad E(\dot{y}_j^2) \geq \sigma_j^2, \quad E(\dot{m}_{ij}^2) \geq \sigma_{ij}^2. \quad (60)$$

Proof. All three inequalities are proved by squaring both sides of the RABAM equations (51)–(53), taking expectations, and using (54) and the fact that the noise is uncorrelated with the additive nonlinear “signal” terms.

Q.E.D.

It is not true that the squared velocity processes are never less than the squared noise processes at every instant. It is only true on average at every instant.

Grossberg's *noise-saturation dilemma* [13] motivates the use of the term "noise suppression" in the RABAM corollary below. The noise-saturation dilemma asks how neurons can have an effective infinite dynamical range when they operate between upper and lower bounds and yet not treat small input signals as noise: "If the x_i are sensitive to large inputs, then why do not small inputs get lost in internal system noise? If the x_i are sensitive to small inputs, then why do they not all saturate at their maximum values in response to large inputs?" [14] This vexing and ubiquitous dilemma, it even confronts the salesperson who tries to balance her presentation between "little" and "big" customers, is the supreme motivator behind Grossberg's shunting-model perspective of neural networks.

Grossberg resolves the saturation half of the dilemma by showing [13], as mentioned above, that shunting models remain sensitive to relative pattern information over a wide range of inputs. He also shows that additive models quickly saturate to upper bounds for large inputs. Indeed this saturation invariance result is arguably Grossberg's greatest achievement. Besides giving information-processing insights into the global dynamics of Hodgkin-Huxley type networks, it also drives Grossberg's conception and implementation of ART behavior, and is at the heart of his recent vision theory. On the other hand, as Carver Mead and other neural VLSI designers have observed, it is well known that a simple logarithmic transduction of local input light intensity into electric potential in the visual system achieves in one stroke both sensitivity to input light intensities over many orders of magnitude and "discounts the illuminant" [14] by equating voltage differences to logarithms of intensity ratios.

Grossberg's resolution of the noise half of the noise-saturation dilemma is far less satisfactory. Grossberg [13] argues that noisy patterns are uniform input patterns and that, for a particular small threshold value, uniform noise is "suppressed" by all neurons in the field shutting off. Besides the dependence on a specific noise threshold, this argument is objectionable on at least two counts. First, noise permeates all parameters and all signals and certainly need not be uniform. Grossberg admits this in his above description of the noise-saturation dilemma when he asks why small inputs do not "get lost in internal system noise." System noise makes everything "jiggle," including relative input pattern values. This is the noise modeled by the additive noise processes in the RABAM equations (51)–(53) or, more realistically, by the additive diffusion processes in the diffusion RABAM equations (47)–(49).

Second, shutting off neurons to suppress noise seems akin to curing the patient by killing him. The goal is to continue "computing" as accurately as possible no matter how noisy the environment. Background noise can be high in feedback systems where noise can multiply by recirculating. In fairness, Grossberg [14] argues that special classes of signal functions, especially sigmoid signal functions, help quench pattern noise by contrast-enhancing input signals. Signal function nonlinearities surely help suppress this special occurrence of noise. But what about synaptic noise? What about joint synaptic and ac-

tivation noise? What about noise compounded by feedback? How do we know such pervasive noise will not prevent an ART system from adaptively resonating, or ruin an adaptive-resonance equilibrium once achieved?

The RABAM noise suppression theorem is an alternative resolution of the noise half of the noise-saturation dilemma. It guarantees that second-order behavior in RABAM systems is as good as it can be: mean-square velocities decrease exponentially quickly to their lower bounds. As the above lemma shows, these lower bounds are just the underlying driving noise variances. Thus the observed fluctuations, the mean-squared velocities, track the unobserved noise fluctuations. Unaided feedback intuitions might easily lead to the prediction that, in light of the lemma, mean-squared velocities may tend toward infinity, especially for widely fluctuating noise processes.

A. RABAM Noise Suppression Theorem

For strictly increasing signal functions S_i and S_j , positive amplification functions a_i , and nondegenerate Hessian conditions: as the RABAM system (51)–(55) converges exponentially quickly, mean-squared activation and synaptic parameters decrease to their lower bounds exponentially quickly:

$$E(\dot{x}_i^2) \downarrow \sigma_i^2, \quad E(\dot{y}_j^2) \downarrow \sigma_j^2, \quad E(\dot{m}_{ij}^2) \downarrow \sigma_{ij}^2. \quad (61)$$

Proof. The proof uses the asymptotic convergence established in the above RABAM theorem for the monotonicity and positivity assumptions and the lower bound on mean-square velocities established in the lemma (60). Then

$$\begin{aligned} \dot{E}(L) &= E(\dot{L}) \\ &= E \left[\sum_i \frac{S'_i \dot{x}_i (n_i - x_i)}{a_i} + \sum_j \frac{S'_j \dot{y}_j (n_j - y_j)}{a_j} \right. \\ &\quad \left. - \sum_i \sum_j \dot{m}_{ij} (n_{ij} - m_{ij}) \right] \end{aligned}$$

by using the positivity of the amplification functions and (51)–(53) to eliminate the terms in braces in (57) in the proof of the RABAM theorem

$$\begin{aligned} &= E \left[\sum_i \frac{S'_i}{a_i} \dot{x}_i^2 - \sum_j \frac{S'_j}{a_j} \dot{y}_j^2 - \sum_i \sum_j \dot{m}_{ij}^2 \right] \\ &\quad + E \left[\sum_i \frac{S'_i}{a_i} \dot{x}_i n_i + \sum_j \frac{S'_j}{a_j} \dot{y}_j n_j + \sum_i \sum_j \dot{m}_{ij} n_{ij} \right] \\ &= E[\dot{L}_{ABAM}] - \sum_i E(n_i) E \left[S'_i \left(b_i - \sum_j S_j m_{ij} \right) \right] \\ &\quad - \sum_j E(n_j) E \left[S'_j \left(b_j - \sum_i S_i m_{ij} \right) \right] \\ &\quad - \sum_i \sum_j E(n_{ij}) E[-m_{ij} + S_i S_j] \\ &\quad + E \left[\sum_i \frac{S'_i}{a_i} n_i^2 + \sum_j \frac{S'_j}{a_j} n_j^2 + \sum_i \sum_j n_{ij}^2 \right] \end{aligned}$$

by using (51)–(53) again to eliminate activation and synaptic velocities in the second expectation above, rearranging, and, as in the proof of the RABAM theorem, using the uncorrelatedness of noise and “signal” terms in (51)–(53) as discussed above to obtain (59)

$$= - \sum_i E \left[\frac{S'_i}{a_i} (\dot{x}_i^2 - n_i^2) \right] - \sum_j E \left[\frac{S'_j}{a_j} (\dot{y}_j^2 - n_j^2) \right] - \sum_i \sum_j (E(\dot{m}_{ij}^2) - \sigma_{ij}^2) \quad (62)$$

by the zero-mean noise assumption (54) and rearrangement. The lemma ensures that the double sum is nonnegative. The RABAM theorem establishes that the Lyapunov function $E(L)$ strictly decreases along trajectories, and thus trajectories end at equilibrium points and arrive there exponentially quickly. This, together with the positivity (and well behavedness [34]) of the weight ratios S'/a , yields the equilibrium conditions:

$$E(\dot{x}_i^2) = \sigma_i^2, \quad E(\dot{y}_j^2) = \sigma_j^2, \quad E(\dot{m}_{ij}^2) = \sigma_{ij}^2. \quad (63)$$

This implies (62).

Q.E.D.

The RABAM noise suppression generalizes the equilibrium conditions obtained in the ABAM theorem in the asymptotic-convergence case. For if the instantaneous “variances” in (63) are zero, then [38] the squared velocities, and thus the velocities, are zero almost everywhere. The zero-variance case is the deterministic case. The sigma-algebra of the probability space is degenerate; it only contains the whole space and the null set. Thus the activation and synaptic velocities are zero everywhere, as in the strict ABAM case. Also note that throughout the proofs of the RABAM theorem and the RABAM noise suppression theorem, the synaptic terms are easier to work with, and the results are “cleaner,” because they do not possess nonlinear signal and amplification terms. We recall again that the above two theorems are also valid for suitably randomized competitive, differential Hebb, and differential competitive learning laws under appropriate conditions.

VII. RABAM ANNEALING AND THE ITO-STRATONOVICH STOCHASTIC CALCULUS

Gradient systems are globally stable. The above theorems are an extension of this general Lyapunov fact. For example, Cohen and Grossberg [6] showed that their symmetric nonlearning autoassociative system can be written in pseudogradient form for monotone increasing signal functions and positive amplification functions.

Geman and Hwang [8] recently showed that stochastic gradient systems with scaled additive Brownian diffusions (noise) perform *simulated annealing* in a weak sense. The gradient is formed from a cost function to be searched by scaled random hill climbing. If the noise is initially scaled high enough (to a physically unrealizable size), then gradually decreasing the nonnegative “temperature” $T(t)$ scaling factor can bounce the system state out of local minima and trap it in global minima. The convergence, though, must proceed exponentially slowly and is only convergence in the weak sense [38] for measures (analogous to the convergence in distribution found in central limit theorems).

The result is not true for convergence with probability one or even convergence probability. There is still some probability that the system state will bounce out of global or near-global minima as “cooling” finishes.

We now extend the RABAM theorem and RABAM noise suppression theorem to include simulated annealing in the general Geman–Hwang sense. For this we introduce the activation “temperatures” or annealing schedules $T_i(t)$ and $T_j(t)$ and the synaptic schedules $T_{ij}(t)$. The temperatures are nonnegative deterministic functions. So they can be brought outside all expectations in proofs. The RABAM annealing model is more general than the Geman–Hwang gradient model, and vastly more general than popular additive-activation annealing models, because learning is permitted and because learning too can be annealed, although perhaps at a different rate than activation annealing. The RABAM annealing model is defined by scaling the diffusion differentials in (47)–(49) with the square root of the corresponding annealing schedules or, in the noise RABAM, by replacing (51)–(53) with (64)–(66):

$$\dot{x}_i = -a_i \left[b_i - \sum_j S_j m_{ij} \right] + \sqrt{T_i} n_i \quad (64)$$

$$\dot{y}_j = -a_j \left[b_j - \sum_i S_i m_{ij} \right] + \sqrt{T_j} n_j \quad (65)$$

$$\dot{m}_{ij} = -m_{ij} + S_i S_j + \sqrt{T_{ij}} n_{ij} \quad (66)$$

where again (67) can be replaced with the other unsupervised learning laws discussed above with appropriate additional constraints.

A. RABAM Annealing Theorem

The RABAM annealing model is globally stable, and asymptotically stable for monotone increasing signal functions and positive amplification functions, in which case the mean-squared activation and synaptic velocities decrease to their temperature-scaled instantaneous “variances” exponentially fast:

$$E(\dot{x}_i^2) \downarrow T_i \sigma_i^2, \quad E(\dot{y}_j^2) \downarrow T_j \sigma_j^2, \quad E(\dot{m}_{ij}^2) \downarrow T_{ij} \sigma_{ij}^2. \quad (67)$$

Proof. The proof largely duplicates the proofs of the RABAM theorem and RABAM noise suppression theorem. Again $E(L)$ is a sufficiently smooth Lyapunov function that allows time differentiation of the integrand. When the diffusion or noise RABAM annealing equations are used to eliminate activation and synaptic velocities in the time-differentiated Lyapunov function, the resulting temperature functions that occur can be factored outside all expectations. The nonnegativity of the temperature functions keeps them from affecting the structure of expanded time derivative of $E(L)$. The random weight functions S'/a are assumed sufficiently well behaved to keep the expectations in which they occur nonnegative. The above lemma is immediately extended to show, for instance, that

the mean-squared velocity $E(\dot{x}_i^2)$ is bounded below by $T_i \sigma_i^2$. Then, (62) is generalized to

$$\begin{aligned} \dot{E}(L) = & - \sum_i E \left[\frac{S'_i}{a_i} (\dot{x}_i^2 - T_i n_i^2) \right] \\ & - \sum_j E \left[\frac{S'_j}{a_j} (\dot{y}_j^2 - T_j n_j^2) \right] \\ & - \sum_i \sum_j (E(m_{ij}^2) - T_{ij} \sigma_{ij}^2). \end{aligned} \quad (68)$$

Q.E.D.

The RABAM annealing theorem is a nonlinear and continuous generalization of Boltzmann machine learning [40], provided learning is Hebbian and very slow. The Boltzmann machine uses discrete symmetric additive autoassociative dynamics. Binary neurons are annealed during periods of Hebbian and anti-Hebbian learning. Here Hebbian learning corresponds to (66) with $T_{ij}(t) = 0$ for all t . Anti-Hebbian learning further replaces the Hebb product $S_i S_j$ in (66) with the negative product $-S_i S_j$. Anti-Hebbian learning (during "free-running" training [40]) can in principle destabilize a RABAM system. This is less likely to occur, though, the slower the anti-Hebbian learning. (The activation terms in the time derivative of $E(L)$ stay negative and can outweigh the possibly positive anti-Hebbian terms, even if learning is fast.) Incidental instability perhaps is not even a problem in this phase of annealing, since the intention is to undo some of the learning in the "environmental" annealing phase. The fundamental distinction between unsupervised RABAM learning and temperature-supervised annealing learning is how noise is treated. Simulated annealing systems search or learn with noise. Unsupervised RABAM systems learn despite noise. During "cooling," the continuous annealing schedules define the flow of RABAM equilibria in the product state space of continuous nonlinear random processes. Equation (67) implies that no finite temperature value, however large, can destabilize a RABAM.

Finally, the proofs of the above RABAM theorems repeatedly use the familiar chain rule of differential calculus. In general, the chain rule does not apply to systems of nonlinear stochastic differential equations, at least not in the general case where each nonlinear parameter is itself a stochastic process. This is the general setting for the Ito calculus. One exception is the related Stratonovich calculus, which defines a stochastic integral (an integral defined with respect to a random measure [41] with as lightly different partitioning of the time interval. The Stratonovich calculus includes the classical chain rule, though in general at the expense of possessing non-Markovian solution processes.

Maybeck [35] shows that, with probability one, the Ito stochastic differential equals the Stratonovich stochastic differential plus a term involving the nonlinear random scaling factor on the underlying Brownian diffusion. The two differentials and corresponding integrals are equal when this extra term is zero. This is fortunately always true for RABAM systems since noise terms are scaled with constants or sequences of constants (deterministic an-

nealing schedules). The extra term involves the derivative of this constant with respect to the corresponding random activation or synapse. Thus RABAM models enjoy the best of both stochastic-calculus worlds. They maintain the familiar chain rule of Stratonovich stochastic dynamical systems and inherit the better-explored properties of Ito stochastic dynamical systems. For instance, all RABAM solution processes are Markov processes. This promises a new approach to nonlinear stochastic optimal estimation and control.

VIII. CONCLUSIONS

The RABAM model unifies many popular feedforward and feedback unsupervised learning systems and extends them to the more realistic, and more complex, random process domain. Unsupervised learning is structurally stable for wide families of nonlinear feedback dynamical systems. This holds for the popular signal Hebb and competitive learning feedback systems under quite general conditions. It holds to a lesser extent for the largely unexplored signal-velocity learning feedback systems that adapt with differential Hebb or differential competitive laws. Pulse-coded [10], [11] signal functions augment the class of feedback systems that can stably learn with the differential Hebb and differential competitive laws, since in this case they give back, respectively, signal Hebb and competitive learning behavior much of the time. The pulse-coding framework also promises new engineering approaches to implementing adaptive networks, perhaps with sinusoidal techniques, as well as suggesting new roles for signal-velocity synaptic mechanisms in real neural systems. The feedback in these stable dynamical systems can always be eliminated to produce unsupervised feedforward systems that stably learn with Hebbian, competitive, or signal-velocity learning laws.

The stability of RABAM models yields the structural stability of ABAM models. From an engineering perspective, this means we can more confidently build large-scale ABAM networks with electrical, optical, electrooptic, and perhaps other (molecular, fluid, plasma, polymer, etc.) devices.

For the neurobiologist, the structural stability of ABAM models suggests that at least some of the consistent criticism that neural models are "unrealistic" is unfounded. The many intricate neuronal and molecular properties that the neurobiologist studies, and finds missing in neural network models, are modeled in RABAM systems as random unmodeled effects. The RABAM noise suppression theorem says these unmodeled effects are ignored by the network's global computations almost as quickly as they are encountered. Like many quantum-level effects in electrical devices, these unmodeled effects simply do not affect the structure of global network computations—so long as they are net random effects.

How plausible is this? Some unmodeled effects of course depend on neuronal and synaptic behavior and so are not accurately modeled as independent noise processes, though perhaps central-limit (Gaussian) effects emerge from the interaction of many such processes. Many correlated effects can also be incorporated as slowly

varying parameters in the "signal" part of the RABAM model.

In general, the sheer number (sample size) of unmodeled effects suggests a Brownian approximation. To the extent that the unmodeled synaptic and neuronal effects involve many independently interacting continuous phenomena, the net result is a Brownian diffusion, as assumed by RABAM models. This is because finite-variance continuous processes with independent increments in time have Gaussian increments [35], and hence give rise to a Brownian diffusion.

ACKNOWLEDGMENT

The author thanks F. Watkins for many valuable technical discussions and conjecture-testing simulations.

REFERENCES

- [1] J. A. Anderson, J. W. Silverstein, S. R. Ritz, and R. S. Jones, "Distinctive features, categorical perception, and probability learning: Some applications of a neural model," *Psycholog. Rev.*, vol. 84, pp. 413-451, 1977.
- [2] J. A. Anderson, "Cognitive and psychological computation with neural models," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 799-815, Sept./Oct. 1983.
- [3] G. A. Carpenter, M. A. Cohen, and S. Grossberg, "Computing with neural networks," *Science*, vol. 235, pp. 1226-1227, Mar. 6, 1987.
- [4] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Comput. Vision, Graph., Image Process.*, vol. 37, pp. 54-115, 1987.
- [5] G. A. Carpenter and S. Grossberg, "ART 2: Self-organization of stable category recognition codes for analog input patterns," *Appl. Opt.*, vol. 26, no. 23, pp. 4919-4930, Dec. 1, 1987.
- [6] 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.*, vol. SMC-13, pp. 815-826, Sept./Oct. 1983.
- [7] M. A. Cohen and S. Grossberg, "Masking fields: A massively parallel neural architecture for learning, recognizing, and predicting multiple groupings of patterned data," *Appl. Opt.*, vol. 26, p. 1866, 1987.
- [8] S. Geman and C. Hwang, "Diffusions for global optimization," *SIAM J. Contr. Optimiz.*, vol. 24, no. 5, pp. 1031-1043, Sept. 1986.
- [9] R. Gilmore, *Catastrophe Theory for Scientists and Engineers*. New York: Wiley, 1981.
- [10] M. A. Gluck, D. B. Parker, and E. Reifsnider, "Some biological implications of a differential-Hebbian learning rule," *Psychobiology*, vol. 16, no. 3, pp. 298-302, 1988.
- [11] M. A. Gluck, D. B. Parker, and E. S. Reifsnider, "Learning with temporal derivatives in pulse-coded neuronal systems," in *Proc. Nov. 1988 IEEE Neural Inform. Proc. Syst. (NIPS) Conf.* (Denver, CO) San Mateo, CA: Morgan Kaufman, in press.
- [12] S. Grossberg, "On learning and energy-entropy dependence in recurrent and nonrecurrent signed networks," *J. Statistic. Phys.*, vol. 1, pp. 319-350, 1969.
- [13] S. Grossberg, *Studies of Mind and Brain*. Boston, MA: Reidel, 1982.
- [14] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures," *Neural Networks*, vol. 1, no. 1, pp. 17-61, 1988.
- [15] R. Hecht-Nielsen, "Counterpropagation networks," *Appl. Opt.*, vol. 26, no. 3, pp. 4979-4984, Dec. 1, 1987.
- [16] M. W. Hirsch and S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*. New York: Academic, 1974.
- [17] M. W. Hirsch, *Differential Topology*. New York: Springer-Verlag, 1976.
- [18] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiology*, vol. 117, pp. 500-544, 1952.
- [19] J. J. Hopfield, "Neural networks with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci., U.S.*, vol. 81, pp. 3088-3092, 1984.
- [20] J. M. Kinser, H. J. Caulfield, and J. Shamir, "Design for a massive all-optical bidirectional associative memory: The big BAM," *Appl. Opt.*, vol. 27, no. 16, pp. 3442-3443, Aug. 15, 1988.
- [21] A. H. Klopff, "A drive-reinforcement model of single neuron function: An alternative to the Hebbian neuronal model," *Proc. Amer. Inst. Phys.: Neural Networks for Computing*, pp. 265-270, Apr. 1986.
- [22] A. H. Klopff, "Drive-reinforcement learning: A real-time learning mechanism for unsupervised learning," in *Proc. IEEE Int. Conf. Neural Networks (ICNN-87)*, vol. II, June 1987, pp. 441-445.
- [23] A. H. Klopff, "A neuronal model of classical conditioning," *Psychobiology*, vol. 16, no. 2, pp. 85-125, 1988.
- [24] T. Kohonen, *Self-Organization and Associative Memory*, second ed. New York: Springer-Verlag, 1988.
- [25] B. Kosko, "Adaptive inference," monograph (Verac, Inc., tech. rep.), June 1985.
- [26] B. Kosko and J. S. Limm, "Vision as causal activation and association," *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 579, pp. 104-109, Sept. 1985.
- [27] B. Kosko, "Differential Hebbian learning," *Proc. Amer. Inst. Phys. Neural Networks for Computing*, pp. 277-282, Apr. 1986.
- [28] B. Kosko and C. C. Guest, "Optical bidirectional associative memories," *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 758, p. 11, 1987.
- [29] B. Kosko, "Competitive adaptive bidirectional associative memories," in *Proc. IEEE Int. Conf. Neural Networks (ICNN-87)*, vol. II, June 1987, pp. 261-268.
- [30] B. Kosko, "Adaptive bidirectional associative memories," *Appl. Opt.*, vol. 26, no. 23, pp. 4947-4960, Dec. 1, 1987.
- [31] B. Kosko, "Bidirectional associative memories," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-18, pp. 49-60, Jan./Feb. 1988.
- [32] B. Kosko, "Feedback stability and unsupervised learning," *Proc. 2nd IEEE Int. Conf. Neural Networks (ICNN-88)*, vol. I, July 1988, pp. 141-152.
- [33] B. Kosko, "Hidden patterns in combined and adaptive knowledge networks," *Int. J. Approx. Reason.*, vol. 2, no. 4, pp. 377-393, Oct. 1988.
- [34] B. Kosko, "Structural stability of unsupervised learning in feedback networks," submitted for publication.
- [35] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, vol. 2. New York: Academic, 1982.
- [36] T. S. Parker and L. O. Chua, "Chaos: A tutorial for engineers," *Proc. IEEE*, vol. 75, no. 8, pp. 982-1008, Aug. 1987.
- [37] W. K. Pratt, *Digital Image Processing*. New York: Wiley, 1978.
- [38] W. Rudin, *Real and Complex Analysis*, second ed. New York: McGraw-Hill, 1974.
- [39] D. E. Rumelhart and D. Zipser, "Feature discovery by competitive learning," *Cognitive Science*, vol. 9, pp. 75-112, 1985.
- [40] D. E. Rumelhart, G. E. Hinton, and J. L. McClelland, "A general framework for parallel distributed processing," in *Parallel Distributed Processing*, vol. 1, D. E. Rumelhart, J. L. McClelland, Eds. Cambridge, MA: M.I.T. Press, 1986, ch. 2.
- [41] A. V. Skorokhod, *Studies in the Theory of Random Processes*. Reading, MA: Addison-Wesley, 1965.
- [42] R. Thom, *Structural Stability and Morphogenesis*. Reading, MA: Addison-Wesley, 1975.
- [43] F. Watkins, personal communication.

*



Bart Kosko (M'85) received the bachelor's degrees in philosophy and economics from the University of Southern California in 1982, the master's degree in applied mathematics from the University of California, San Diego, in 1983, and the Ph.D. degree in electrical engineering from the University of California, Irvine, in 1987.

He is an Assistant Professor of Electrical Engineering at the University of Southern California. He is also Managing Editor of the Springer-Verlag monograph series *Lecture Notes in Neural*

Computing and is Associate Editor of *Neural Networks*, *IEEE TRANSACTIONS ON NEURAL NETWORKS*, *Journal of Mathematical Biology*, and *Lecture Notes in Biomathematics*.

Dr. Kosko is an elected governor of the Internal Neural Network Society and is a USC Shell Oil Faculty Fellow. He is Program Co-chairman of the Summer 1990 International Joint Conference on Neural Networks. He was Program Chairman of the 1988 IEEE International Conference on Neural Networks (ICNN-88) and Program and Organizing Chairman of the first IEEE ICNN-87.