PHYSICAL REVIEW E VOLUME 47, NUMBER 6 JUNE 1993

Analog neural networks with local competition. II. Application to associative memory

F. R. Waugh and R. M. Westervelt
Division of Applied Sciences and Department of Physics, Harvard University, Cambridge, Massachusetts 02138

(Received 18 September 1992; revised manuscript received 25 January 1993)

We analyze the attractors of associative-memory neural networks in which analog neurons compete locally. These networks are well suited for a variety of feature-extraction, pattern-classification, and data-compression tasks. For networks storing a finite number of patterns, we present bifurcation diagrams for the pattern overlaps. For networks storing an extensive number of patterns, we present phase diagrams showing attractor types as a function of pattern-storage fraction and neuron-transfer-function steepness. We also report results for the storage capacity of k-winner associative memories in the limit of infinite neuron gain. Numerical investigations of computer-generated networks confirm the phase diagrams.

PACS number(s): 87.10.+e, 06.50.Mk, 64.60.Cn

I. INTRODUCTION

Associative memory has long been a benchmark for measuring the performance of novel neural network architectures [1–6]. Over the last decade, statistical techniques have been developed to analyze associative-memory storage [7,8] and applied with great success to fully connected, stochastic networks of two-state neurons [9]. It is important now to apply these techniques to network architectures that can solve real-world problems using readily available technology.

In this paper, we study associative memory in networks in which analog-valued neurons complete in localized clusters. We introduced these networks and discussed their stability properties in the preceding paper (Ref. [10], hereinafter referred to as I). Localized competition makes these networks well suited for feature-extraction or pattern-classification applications, in which input data must be assigned to one or a few of many different categories. Such problems occur frequently in image processing, where the different categories might be different colors, depths, textures, elementary image features, or written characters. Other possible applications include speech recognition [11,12], analysis of DNA, proteins, and other complex molecules [13–15], and data-compression methods such as principal-component analysis and vector quantization [11,16].

The architecture of competitive analog networks is shown schematically in Fig. 1. The networks are similar to standard analog networks [17–28], in that a set of deterministic update equations determines each neuron's output by passing its input through a smooth, continuous analog transfer function. The update equations, in which time can be either a continuous or discrete variable, describe networks of resistively coupled nonlinear amplifiers that either run freely or are clocked externally. Such networks can be easily implemented in analog electronic circuitry [10]. Benefits of analog processing include discrete-time, parallel updating without oscillation [10,20–22,29] and suppression of spurious fixed-point attractors [24,25].

What distinguishes competitive analog networks is that, in addition to communicating through synaptic interconnections, neurons also compete through a mechanism that constrains neuron outputs in localized clusters to sum to a constant at all times. Competition makes the output of a neuron depend on the inputs of all neurons in its cluster, enabling neurons to perform more complicated computations than are possible in standard analog networks. We concentrate here on clusters that implement winner-take-all or k-winner functions of their inputs, meaning that one or k neurons have large outputs in a

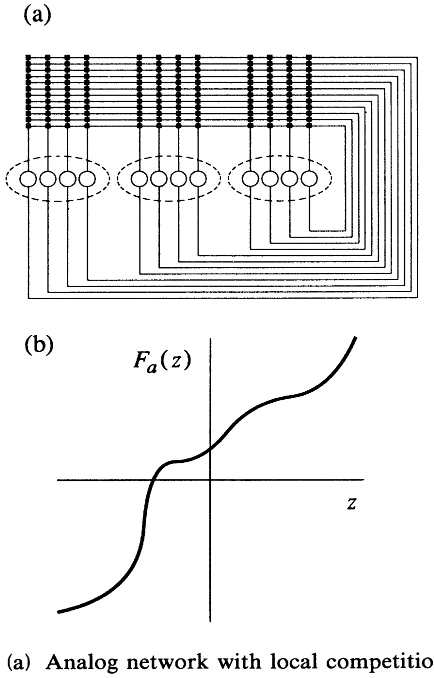


FIG. 1. (a) Analog network with local competition. Circles represent neurons, filled squares represent interconnections, and dashed ellipses demark clusters of competing neurons. Network shown has $N=3$ clusters with $Q=4$ neurons per cluster. (b) Smooth, continuous analog neuron transfer function that satisfies conditions given in Sec. II of paper I.

1063-651X/93/47(6)/4537(15)/$06.00

cluster at any time while the other neuron outputs are suppressed. Other network architectures with similar competitive interactions include the neocognitron [30], a variety of classifiers and vector quantizers [11,16,31,32] and networks of Potts spins [33–42].

The networks we study are configured as associative memories in which some subset of neurons in each cluster are chosen to win the competition in each stored pattern. The main result of this paper is a set of analytical phase diagrams [7,23,26,27] describing the attractors of these networks as a function of the neuron-transfer-function steepness and the ratio of the number of stored patterns to the number of clusters. By indicating regions of parameter space where memory recall is possible as well as regions where spurious fixed points or oscillatory attractors exist, the phase diagrams provide quantitative guidelines for designing and operating associative memories. A similar diagram has been reported recently for an associative memory of Potts spins [38,39]. Similarities and differences between competitive and Potts networks are noted throughout the paper. From a practical viewpoint, competitive networks are much more easily implemented in electronic circuitry than are Potts networks.

In Sec. II, we discuss how to configure analog competitive networks as associative memories. We outline in Sec. III the statistical techniques we use to study memory storage and retrieval in these networks, and we define the different types of attractors—paramagnetic, recall, spin-glass, and oscillatory—the networks can have. We analyze the retrieval capability of competitive networks that store a finite number of patterns in Sec. IV and an extensive number of patterns in Sec. V. The results of this analysis are summarized in bifurcation diagrams for the memory overlap in finitely loaded networks (Sec. IV) and phase diagrams for several different configurations of extensively loaded networks (Sec. V). In Sec. VI, we report storage capacities for k-winner networks in the limit of infinite neuron gain. Section VII contains results of numerical investigations that support the analytical results. Our results are summarized in Sec. VIII.

II. COMPETITIVE ASSOCIATIVE MEMORIES

In this section, we describe competitive analog networks configured to operate as associative memories. A more detailed explication of competitive networks appears in I.

The networks we study evolve according to either the continuous-time differential equations

$$\frac{dx_{ia}(t)}{dt}=-x_{ia}(t)+F_a(h_{ia}(t)+B_i(t)) \tag{1}$$

or the discrete-time, parallel-update equations

$$x_{ia}(t+1)=F_a(h_{ia}(t)+B_i(t)) , \tag{2}$$

where

$$h_{ia}(t)=\sum_{j=1}^{N}\sum_{b=1}^{Q}J_{ij}^{ab}x_{jb}(t) . \tag{3}$$

In Eqs. (1) and (2), the index i labels the N clusters $(i=1,\ldots,N)$ and the index a labels the Q neurons in each cluster i $(a=1,\ldots,Q)$. The neuron inputs $h_{ia}(t)$, the neuron outputs $x_{ia}(t)$, the analog input-output transfer functions $F_a(z)$, and the interconnection matrix J_{ij}^{ab} are all real valued. The time-dependent bias terms $B_i(t)$ are determined implicitly by the competitive constraints [10]

$$\sum_{a=1}^{Q}x_{ia}(t)=0 \quad \text{(continuous time)} , \tag{4}$$

$$\sum_{a=1}^{Q}x_{ia}(t+1)=0 \quad \text{(discrete time)} . \tag{5}$$

The Q transfer functions $F_a(z)$, $a=1,\ldots,Q$, are the same for each cluster. Thus the clusters all have the same cluster gain β [10]. The transfer functions are normalized so that outputs of winning neurons are approximately $1/k$. To ensure that all neuron outputs x_{ia} equal zero for sufficiently low cluster gain, we require that all transfer functions have the same value of $F_a(0)$. We pay particular attention to networks in which all transfer functions are given either by

$$F_a(z)=\frac{1}{Q-1}[Q\exp(\gamma z)-1] , \tag{6}$$

which we refer to as winner-take-all networks, or by

$$F_a(z)=\frac{1}{k(Q-k)}\{Q[1+\exp(-\gamma z)]^{-1}-k\} , \quad 1\le k\le Q-1 , \tag{7}$$

which we refer to as k-winner networks. In winner-take-all networks, the neuron with the largest input in each cluster has a large output, while the other $Q-1$ neuron outputs are suppressed. In k-winner networks, the neurons with the k largest inputs in each cluster have large outputs, while the other $Q-k$ neuron outputs are suppressed. Examples of the transfer functions (6) and (7) are depicted in Fig. 2. The parameter γ, which we refer to as the *neuron gain,* controls the slope of both functions.

The interconnection matrix J_{ij}^{ab} couples the output of neuron b in cluster j to the input of neuron a in cluster i. The matrix is constructed to store p patterns ξ_i^μ, where $\mu=1,\ldots,p$ and $i=1,\ldots,N$, using a form of the Hebb rule [33]:

$$J_{ij}^{ab}=\frac{1}{NQ^2}\sum_{\mu=1}^{p}(Q\delta_{a,\xi_i^\mu}-k)(Q\delta_{b,\xi_j^\mu}-k) , \quad J_{ii}^{ab}=0 . \tag{8}$$

Examples of patterns are shown in Fig. 3. Each ξ_i^μ is a set of different integers indicating which neurons are chosen to win the competition in cluster i for pattern μ. The k different integers in this set, where $1\le k\le Q-1$, are chosen randomly and without bias from the set of integers $\{1,\ldots,Q\}$. For example, the statement $\xi_i^\mu=\{1,2,\ldots,k\}$ means that, in pattern μ, neurons $1,2,\ldots,k$ are chosen to win the competition in cluster i. For winner-take-all networks, the value of k in Eq. (8) is always 1, while for k-winner networks, it is in the range $1\le k\le Q-1$. For both network types, the case $Q=2$ is equivalent to associative memories of standard analog

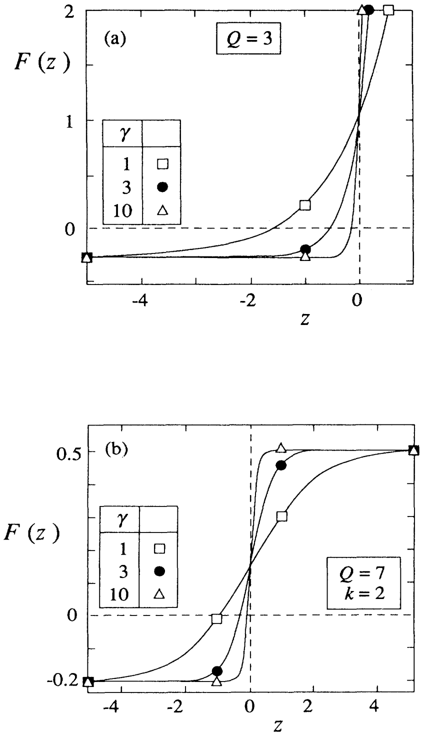
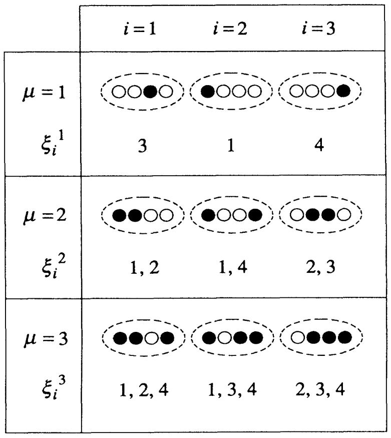


FIG. 2. Neuron transfer functions used in competitive associative memories: (a) winner-take-all functions, Eq. (6), for cluster of $Q=3$ neurons with $\gamma=1$, 3, and 10; (b) k-winner functions, Eq. (7), for cluster of $Q=7$ neurons and $k=2$ winners with $\gamma=1$, 3, and 10.

neurons [23–28]. Sums over the patterns ξ_i^μ in Eq. (8) and the rest of this paper imply summation over all k values of ξ_i^μ. Other forms of the Hebb rule that have been studied for Potts networks [42] are not considered here.



	$i=1$	$i=2$	$i=3$
$\mu=1$	○○●○	●○○○	○○○●
$\xi_i^{\ 1}$	3	1	4
$\mu=2$	●●○○	●○○●	○●●○
$\xi_i^{\ 2}$	1, 2	1, 4	2, 3
$\mu=3$	●●○●	●○●●	○●●●
$\xi_i^{\ 3}$	1, 2, 4	1, 3, 4	2, 3, 4

FIG. 3. Examples of patterns that can be stored in competitive network of Fig. 1(a), showing how ξ_i^μ and k are defined. For each of patterns 1, 2, or 3, $k=1$, 2, or 3 neurons are chosen to win competition in each cluster. Winner-take-all networks can store patterns like ξ^1; k-winner networks can store patterns like ξ^1, ξ^2, or ξ^3.

III. STATISTICAL MECHANICS FOR ANALOG NETWORKS

We use the Liapunov function $L(t)$ [Eq. (18) of I] and the stability criterion [Eq. (39) of I] to analyze the attractors of competitive analog networks. The procedure is to treat $L(t)$ as an energy [27] and to apply standard techniques of statistical mechanics for disordered systems [7,9]. A crucial step in this procedure is to introduce an auxiliary temperature, which has no physical meaning and is set to zero at the end of the calculation [27]. The auxiliary temperature enables the derivation of a free energy f per neuron, which provides information about metastable states. When the auxiliary temperature is set to zero, the metastable states become the fixed points of the dynamical system for which $L(t)$ is a Liapunov function. For a particular interconnection matrix J_{ij}^{ab}, the free energy per neuron is [27]

$$f=-\frac{1}{\tilde{\beta}N}\ln\int_{-\infty}^{\infty}\prod_{i=1}^{N}\prod_{a=1}^{Q}[d\rho(x_{ia})]\exp[-\tilde{\beta}L(t)]\ , \qquad (9)$$

where $\tilde{\beta}$ is the inverse of the auxiliary temperature and $d\rho(x_{ia})$ equals 1 on the range of the transfer function $F_a(z)$ and 0 otherwise [27].

We stress that the networks we study are completely deterministic; the auxiliary temperature is simply a mathematical device allowing the use of statistical mechanics to learn about attractors. Thus we are not interested in the free energy at nonzero values of the auxiliary temperature (which describes networks of *noisy* analog neurons). Note that this procedure works only for networks in which all cluster gains satisfy the stability criterion, since otherwise $L(t)$ is not a Liapunov function. Other methods can be used to analyze networks in the region in which the stability criterion is violated [28,43]. In this paper, however, we consider period-two limit cycles to be undesirable and study only networks guaranteed to be free of them.

Important issues for reliable associative-memory performance include how many patterns can be stored, what other fixed points exist besides those corresponding to stored patterns, how neuron gain affects pattern retrieval, and how the stability criterion appearing as Eq. (39) of I affects discrete-time, parallel-update networks. In the next four sections, we address these issues both analytically, by using the free energy (9) to derive phase diagrams [7,23,26,27] summarizing different attractor types, and numerically, by studying attractors of small computer-generated networks. Useful quantities for characterizing attractors are the p pattern overlaps m_μ, defined as

$$m_\mu\equiv\frac{1}{N}\sum_{i=1}^{N}x_{i\xi_i^\mu}\ ,\quad \mu=1,\ldots,p \qquad (10)$$

and a spin-glass order parameter q, defined in Sec. V as

$$q\equiv\frac{1}{NQ}\frac{k(Q-k)}{(Q-1)}\sum_{i=1}^{N}\sum_{a=1}^{Q}x_{ia}^2\ . \qquad (11)$$

Successful recall of pattern μ means that m_μ is of order 1, while all other overlaps m_ν, $\nu\neq\mu$, are much less than 1.

We show that in the limit of finite memory loading, in which the storage fraction $\alpha \equiv p/N$ is zero, two attractor types can occur: (i) a paramagnetic fixed point, for which $q=0$, and all $m_\mu = 0$, and (ii) memory recall fixed points, for which $q>0$ and one m_μ is of order 1. When the storage fraction α is greater than zero, two other attractor types can exist in addition to the paramagnetic and recall attractors: (iii) spin-glass fixed points, for which $q>0$ but all $m_\mu = 0$, and (iv), period-two limit cycles, which can occur when the stability criterion of I is violated in discrete-time, parallel-update networks.

IV. COMPETITIVE ASSOCIATIVE MEMORIES WITH FINITE MEMORY LOADING

We first consider competitive associative memories in the limit of finite memory loading, in which the number p of stored patterns remains finite while the number N of clusters becomes large, so that $\alpha=0$. The analysis in this limit is particularly simple because the interference between patterns is negligible [7]. We show that a discontinuous, hysteretic transition from paramagnetic to memory recall behavior can occur as neuron gain γ increases.

Figure 4, which is the main result of this section, shows bifurcation diagrams for the overlap m when $m_\mu = m\delta_{\mu,1}$ in winner-take-all and k-winner networks with finite memory loading. The diagrams show the overlap as a function of $\hat{\gamma} \equiv \gamma/Q$ for winner-take-all networks and $\hat{\gamma} \equiv \gamma k(Q-k)/Q^2$ for k-winner networks. For both network types, a single solution exists at low $\hat{\gamma}$ and three solutions exist at high $\hat{\gamma}$. The single solution $m=0$ at low $\hat{\gamma}$ is the paramagnetic solution, for which all neuron outputs x_{ia} equal zero. The three solutions at high $\hat{\gamma}$ are the paramagnetic solution and two recall solutions that approach the values 1 and $-1/(Q-1)$ for winner-take-all networks and 1 and $-1/(\hat{Q}-1)$ for k-winner networks, where $\hat{Q} \equiv \max\{Q/k, Q/(Q-k)\}$.

Solutions for m correspond to fixed points of the update equations (1) and (2) only if they are stable. Stable solutions are indicated by solid curves and unstable solutions by dashed curves in Fig. 4. We demonstrate in Appendix B that (i) the paramagnetic solution is stable for $\hat{\gamma} \le 1$ and unstable for $\hat{\gamma} > 1$; (ii) the positive recall solution is always stable; and (iii) the negative recall solution is unstable for winner-take-all networks with $Q>2$ but can be stable at low gain for k-winner networks.

Two distinct types of behavior appear in Fig. 4. For $Q=2$ in winner-take-all networks and $\hat{Q}=2$ in k-winner networks, a single bifurcation occurs at $\hat{\gamma}=1$, in which the paramagnetic solution becomes unstable and two stable recall solutions appear. Stable paramagnetic and recall solutions never coexist. For all other values of Q and $\hat{Q}$ that we have tested, two bifurcations occur: one at $\hat{\gamma}=1$, in which the paramagnetic solution becomes unstable, and another at $\hat{\gamma}=\hat{\gamma}^* < 1$, in which a stable recall solution and another unstable solution appear at a nonzero value $m=m^*$. Stable paramagnetic and recall solutions coexist for $\hat{\gamma}^* < \hat{\gamma} < 1$, and the networks are hysteretic in this region. The values $\hat{\gamma}^*$ and m^* are shown in the insets of Fig. 4.

The bifurcation diagrams of Fig. 4 are derived from the free energy (9) [7,27,34,36,42]. For finite p and Q, the free energy per neuron averaged over patterns in the limit $N \to \infty$ is

$$f=\frac{1}{2}\sum_{\mu=1}^{p} m_\mu^2-\sum_{\mu=1}^{p}\left\langle m_\mu F_{\xi^\mu}\left[\sum_{\nu=1}^{p} m_\nu \delta_{\xi^\mu,\xi^\nu}+B\right]\right\rangle_\xi + \sum_{a=1}^{Q}\left\langle G_a\left[F_a\left[\sum_{\mu=1}^{p} m_\mu \delta_{a,\xi^\mu}+B\right]\right]\right\rangle_\xi . \qquad (12)$$

In Eq. (12), the brackets $\langle\ \rangle_\xi$ denote an average over all possible realizations of the stored patterns, the functions $G_a(x)$ are given by Eq. (19) of I, and the overlaps m_μ obey the saddle point equations

$$m_\mu=\left\langle F_{\xi^\mu}\left[\sum_{\nu=1}^{p} m_\nu \delta_{\xi^\mu,\xi^\nu}+B\right]\right\rangle_\xi . \qquad (13)$$

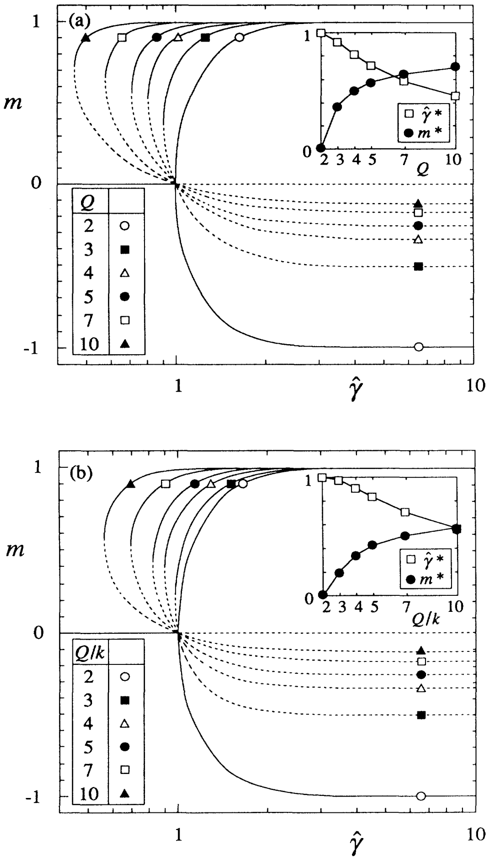


FIG. 4. Bifurcation diagrams for overlap m as a function of gain parameter $\hat{\gamma}$ in networks with finite loading, showing first-order transition from paramagnetic to recall behavior. Curves are for $Q=2$, 3, 4, 5, 7, and 10 neurons per cluster in (a) winner-take-all networks (for which $\hat{\gamma}=\gamma/Q$) and (b) k-winner networks with $k=1$, or $Q-1$ [for which $\hat{\gamma}=\gamma k(Q-k)/Q^2$]. Stable solutions are indicated by solid curves, unstable solutions by dashed curves. For k-winner networks, solutions for m depend only on Q/k; however, stability of negative solution is Q dependent. Insets show values $\hat{\gamma}^*$ of gain parameter and m^* of overlap when the nonzero solution for overlap first appears.

The quantity B in Eqs. (12) and (13) is determined implicitly by the competitive constraint

$$\sum_{a=1}^{Q}\left\langle F_a\left[\sum_{\mu=1}^{p} m_\mu \delta_{a,\xi^\mu}+B\right]\right\rangle_{\xi}=0 . \tag{14}$$

The derivation of Eqs. (12)–(14) is outlined in Appendix A.

To construct bifurcation diagrams, we look for solutions of Eqs. (12)–(14) with $m_\mu = m\delta_{\mu,1}$ in networks in which all neurons have the same transfer function $F_a(z)=F(z)$. Inserting these simplifications and performing the average over patterns leads to

$$f=\tfrac{1}{2}m^2-mF(m+B)+kG(F(m+B)) \\ +(Q-k)G(F(B)) , \tag{15}$$

$$m=kF(m+B) , \tag{16}$$

$$kF(m+B)+(Q-k)F(B)=0 . \tag{17}$$

Combining Eqs. (16) and (17) gives a self-consistent equation for the overlap m:

$$m=-(Q-k)F\left[F^{-1}\left[\frac{m}{k}\right]-m\right] . \tag{18}$$

This result is analogous to the equation $m=F(m)$ which holds for standard analog associative memories at finite loading [27]. For winner-take-all and k-winner networks, Eq. (18) reads

$$m=\begin{cases} 1-[(Q-1)m+1]\exp(-\gamma m) & \text{(winner-take-all)} \quad (19)\\ 1-\dfrac{Q}{k}\left[1+\dfrac{(Q/k-1)(1-m)}{(Q/k-1)m+1}\exp(\gamma m)\right]^{-1} & (k\text{-winner}) . \quad (20)\end{cases}$$

Equation (19) is identical to the mean-field equation describing Potts associative memories at temperature $T=Q(Q-1)/\gamma$ in the limit of finite loading [34,36,42]. However, as we show below, this equivalence does not persist for extensive loading [27,28].

Figure 5 shows typical behavior of the mean-field equations (19) and (20). In Figs. 5(a)–5(c), the left-hand and right-hand sides of the winner-take-all mean-field equation (19) are plotted as functions of m for three different values of $\hat{\gamma}$ for $Q=7$. The three values of $\hat{\gamma}$ are chosen to be less than $\hat{\gamma}^*$, between $\hat{\gamma}^*$ and 1, and greater than 1. Figures 5(d)–5(f) show similar plots for the k-winner mean-field equation (20) for $Q=7$ and $k=1$ or 6. Two invariance properties of Eq. (20) are apparent in Fig. 5. First, because Q and k appear in Eq. (20) only in the ratio Q/k, any solution for k winners in Q-neuron clusters is also a solution for nk winners in nQ-neuron clusters, for all positive integers n. Second, the solutions of Eq. (20) for k and $(Q-k)$ winners per cluster are identical, despite the fact that, as seen in the figure, the right-hand side of Eq. (20) is not invariant under the transformation $k\rightarrow(Q-k)$. Thus solutions of Eq. (20) with the same value of $\hat{Q}\equiv\max[Q/k,Q/(Q-k)]$ are identical.

Stability of solutions of the mean-field equations is determined by the eigenvalues of the matrix $\partial^2 f/\partial m_\rho \partial m_\sigma$ of second derivatives of the free energy with respect to the overlaps. Solutions are stable only if all eigenvalues are positive. The eigenvalues are calculated in Appendix B; for the case $m_\mu = m\delta_{\mu,1}$ of a single successfully recalled pattern, the matrix is diagonal, with two eigenvalues. As shown in Fig. 4, the positive recall solution is always stable for winner-take-all and k-winner networks, while the negative recall solution is always unstable for winner-take-all networks with $Q>2$ and for k-winner networks with $Q>2$ and $k=1$ or $Q-1$. However, the negative recall solution can be stable at low gain for k-winner networks with $1<k<Q-1$. This result is illustrated in Fig. 6, which shows, for various k, the value of $\hat{\gamma}$ at which the negative recall solution becomes unsta-

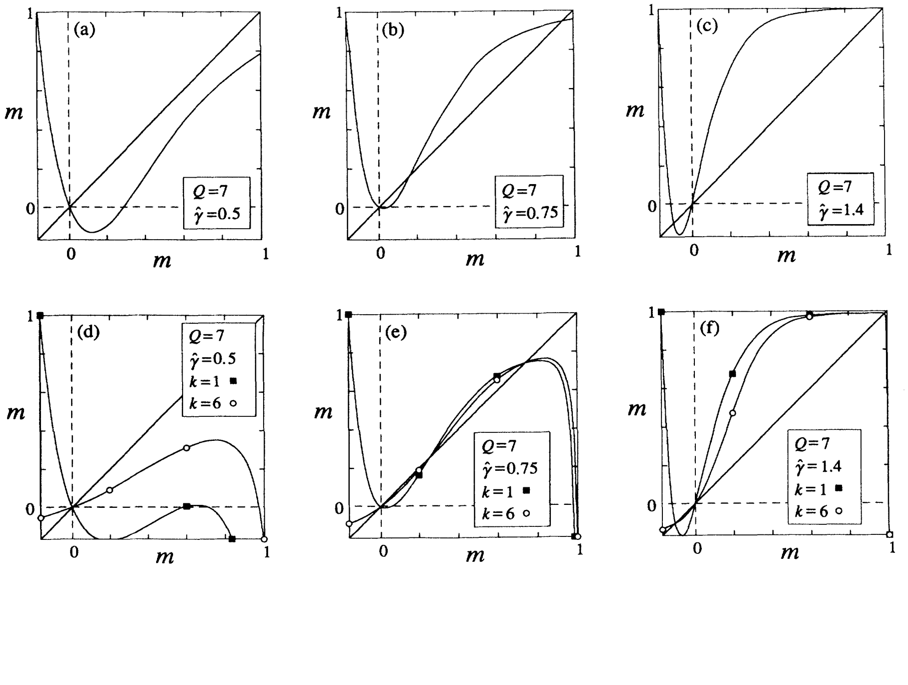


FIG. 5. Left-hand side (diagonals) and right-hand side (curves) of mean-field equations (19) and (20) for finite loading as functions of overlap m. Intersections of curves and diagonals determine bifurcation diagrams appearing in Fig. 4. (a)–(c) Winner-take-all networks, Eq. (19), with $Q=7$, and $\hat{\gamma}=0.5$, 0.75, and 1.4; (d)–(f) k-winner networks, Eq. (20), with $Q=7$, $k=1$ or 6, and $\hat{\gamma}=0.5$, 0.75, and 1.4.

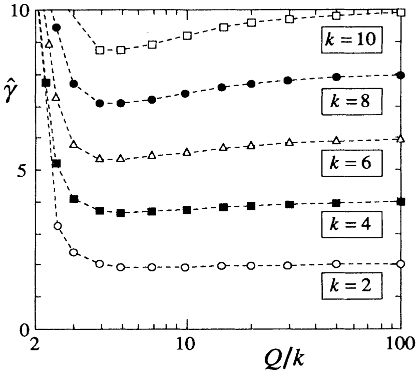


FIG. 6. Values of $\hat{\gamma}$ at which negative recall solution becomes unstable in finitely loaded networks as a function of Q/k for $k=2$, 4, 6, 8, and 10. Stability is determined by eigenvalues of $\partial^2 f/\partial m_\rho \partial m_\sigma$.

ble as a function of Q/k.

Finally, we note that spin-glass and oscillatory attractors do not appear in networks with finite memory loading. Spin-glass attractors do not appear because, as implied by Eq. (13), the overlaps are nonzero whenever the neuron outputs are nonzero. Oscillatory attractors do not appear because, as is shown in Sec. V, the interconnection matrix (8) is positive definite for finite loading so that the stability criterion is satisfied for all values of cluster gain. Thus paramagnetic and recall attractors are the only attractor types.

We have shown in this section that competitive associative memories with finite memory loading undergo a discontinuous, hysteretic transition from paramagnetic to recall behavior as their transfer functions steepen. This result yields the $\alpha=0$ axes of the phase diagrams that appear in the next section for extensively loaded networks. We show in the next section that the hysteretic transition from paramagnetic to recall behavior persists at low but finite α. This is in contrast to standard analog networks, for which the phase diagram exhibits a spin-glass region between the paramagnetic and recall regions for all $\alpha>0$ [23,26,27]. We have also seen that, in the finite loading limit, competitive winner-take-all associative memories and finite-temperature Potts associative memories obey the same mean-field equation. This equivalence does *not* persist at finite α, due to differences between deterministic and stochastic dynamics.

V. COMPETITIVE ASSOCIATIVE MEMORIES WITH EXTENSIVE MEMORY LOADING

We now consider competitive associative memories in which the number p of patterns varies extensively with the number N of clusters as $p=\alpha N$. The interference between patterns can no longer be ignored in these networks [7]. We use the replica method, assuming replica symmetry, to derive a set of self-consistent equations for the overlaps m_μ and the spin-glass order parameter q. We expect replica-symmetry-breaking effects to be small, as they typically are for Hebb-rule associative memories [44,45].

Figures 7 and 8 show several analytical phase diagrams [7,23,26,27] for continuous-time and discrete-time networks with transfer functions given by Eqs. (6) and (7). These diagrams, which are the main result of this paper, indicate the types of attractors that the networks can have as a function of the gain parameter $\hat{\gamma}$ and the ratio $\alpha=p/N$ of patterns to clusters. [Recall that $\hat{\gamma}\equiv\gamma/Q$ for winner-take-all networks and that $\hat{\gamma}\equiv\gamma k(Q-k)/Q^2$ for k-winner networks.] The diagrams are valid in the limit of large N and finite Q.

The diagrams of Fig. 7, which are for continuous-time updating, each contain three regions labeled pm (paramagnetic), recall, and sg (spin glass). In the paramagnetic region, the networks have a single global attractor at the origin of state space, $x_{ia}=0$ for all a and i. Thus $q=0$ and all $m_\mu=0$ in this region. In the spin-glass region, the networks have many fixed-point attractors away from the origin. These attractors are characterized by $q>0$ and all $m_\mu=0$: the paramagnetic attractor is unstable, but fixed points corresponding to stored patterns have not yet appeared. In the recall region, the networks function reliably as associative memories, with fixed points that have large overlaps with the stored patterns. These fixed points are characterized by $q>0$ and one or more $m_m>0$.

The diagrams of Fig. 8, which are for discrete-time, parallel updating, also contain paramagnetic, recall, and spin-glass regions. In addition, each contains a region marked osc (oscillatory) in which the stability criterion [Eq. (39) of I] is violated. The function $L(t)$ [Eq. (18) of I] is not a Liapunov function in this region. Recall and spin-glass attractors may still exist, but the networks can also have period-two limit cycle attractors. Outside the oscillation region, the phase diagrams are identical to those for continuous-time updating. Although no oscillatory regions appear in the phase diagrams of continuous-time networks, unavoidable neural and synaptic delays can lead to oscillatory attractors in electronic implementations [46].

In Figs. 7(a)–7(c) and 8(a)–8(c), the recall region is itself divided into two parts by a dashed curve. In the smaller, low-gain part, recall fixed points coexist with the paramagnetic fixed point, while in the larger, high-gain part, they coexist with spin-glass fixed points. Similar behavior has been reported in stochastic Potts associative memories for the case $Q=3$ [38,39]. The effect is important because it implies that the spurious attractors that often degrade associative-memory performance [24,25] are not present in the low-gain part of the recall region. As seen in Figs. 7 and 8, the effect is more prominent at higher values of Q. In Figs. 7(d) and 8(d), recall fixed points coexist only with spin-glass fixed points.

The phase diagrams are computed from the free energy (9) and, for discrete-time updating, from the stability criterion [Eq. (39) of I]; details of the calculation appear in Appendix C. When a network has nonzero overlaps m_ν with finite number s of patterns, $\nu=1,\ldots,s$, the free energy f per neuron is determined in the limit of large N as a saddle point over the overlaps m_ν, the spin-glass order parameter q, and a third quantity C. The saddle-point equations are

$$m_\nu = \langle \hat{x}_{\xi^\nu} \rangle_{z,\xi}, \quad \nu = 1, \dots, s \ , \tag{21}$$

$$q = \frac{1}{Q}\frac{k(Q-k)}{Q-1}\left\langle \sum_{a=1}^{Q} \hat{x}_a^2 \right\rangle_{z,\xi} , \tag{22}$$

$$C = \left[\frac{1}{\alpha r Q}\frac{k(Q-k)}{Q-1}\right]^{1/2}\left\langle \sum_{a=1}^{Q} z_a \hat{x}_a \right\rangle_{z,\xi} . \tag{23}$$

In Eqs. (21)–(23), the Q quantities $\hat{x}_a$ are determined implicitly by

$$\hat{x}_a = F_a\left[\sum_{\nu=1}^{s} m_\nu \delta_{a,\xi^\nu} + z_a\left[\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}\right]^{1/2} + \hat{x}_a\frac{\alpha}{Q}\frac{k(Q-k)}{Q-1}(\tilde{r}-1) + B\right] , \tag{24}$$

with B determined by the competitive requirement $\sum_a \hat{x}_a = 0$. The brackets $\langle\ \rangle_{z,\xi}$ indicate an average over the patterns ξ^ν and over the Q continuous variables z_a using a Gaussian distribution:

$$\langle\ \rangle_{z,\xi} \to \left\langle \int_{-\infty}^{\infty} \prod_{a=1}^{Q}\left[\frac{dz_a}{\sqrt{2\pi}}\right]\exp\left[-\frac{1}{2}\sum_a z_a^2\right](\ )\right\rangle_{\xi} . \tag{25}$$

The quantities r and $\tilde{r}$ are

$$r = \frac{q}{(1-C)^2} , \quad \tilde{r} = \frac{1}{1-C} . \tag{26}$$

The boundaries in Figs. 7 and 8 are calculated as follows. The boundary of the recall region, also known as the storage capacity α_c, is found numerically as the largest value of α for which a solution of Eqs. (21)–(23) exists with $m_1 \sim 1$ and all other $m_\nu = 0$. The boundary between the spin-glass and origin regions is found by expanding Eqs. (21)–(23) to leading order in the small quantities $\hat{x}_a$. This expansion is carried out in Appendix D. For networks in which all neurons have the same transfer function $F(z)$, the boundary has the analytical form

$$F'(F^{-1}(0)) = \frac{Q}{k(Q-k)}\frac{1}{1+2\sqrt{\alpha/(Q-1)}} , \tag{27}$$

where $F'(z)$ is the derivative of $F(z)$. For both winner-take-all and k-winner networks, Eq. (27) yields

$$\hat{\gamma} = \frac{1}{1+2\sqrt{\alpha/(Q-1)}} . \tag{28}$$

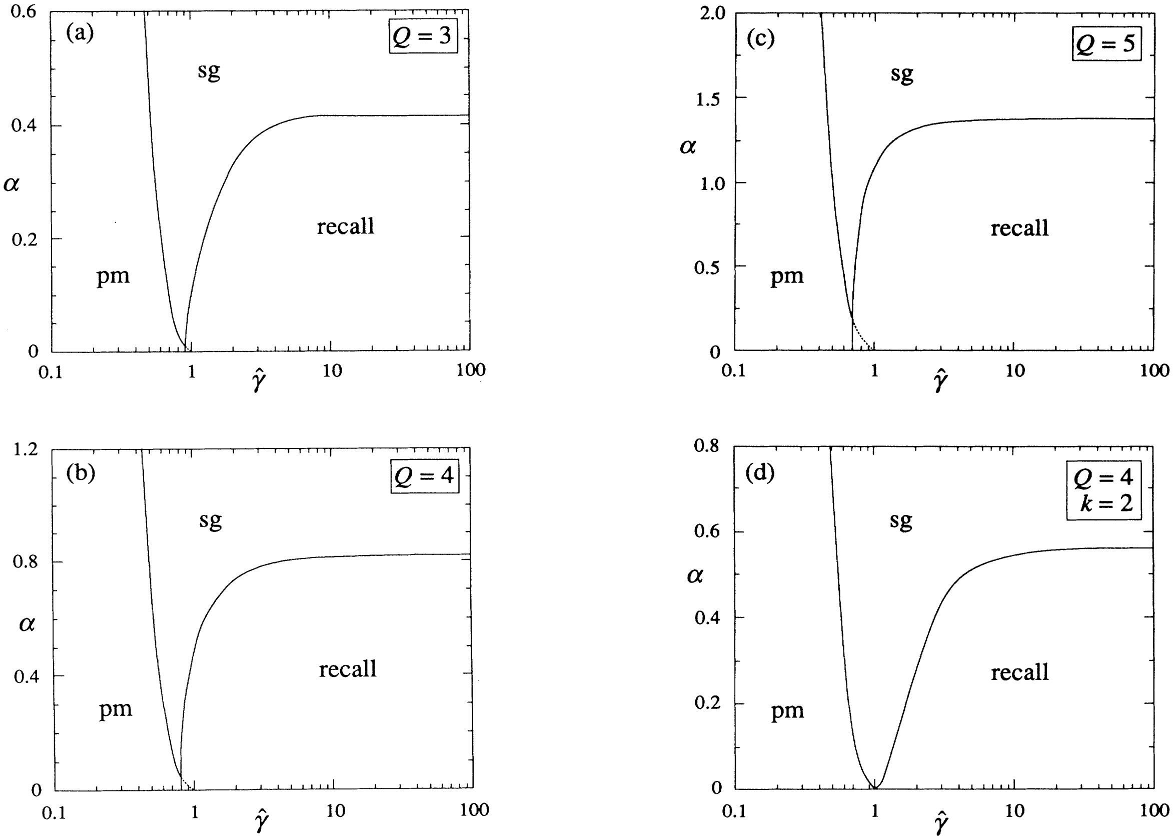


FIG. 7. Phase diagrams for extensively loaded associative memories with continuous-time updating, showing different attractor types as function of storage fraction α and gain parameter $\hat{\gamma}$. First three diagrams are for winner-take-all networks with (a) $Q=3$, (b) $Q=4$, and (c) $Q=5$ neurons per cluster; fourth diagram (d) is for k-winner networks with $Q=4$ neurons and $k=2$ winners per cluster. Labels pm, sg, and recall denote paramagnetic, spin-glass, and recall regions. Dashed curves within recall region in (a)–(c) indicate boundary between coexistence of recall attractors with paramagnetic attractor (low gain) and with spin-glass attractors (high gain).

Finally, the boundary of the oscillation regions of Fig. 8 is determined by the stability criterion derived in I. For winner-take-all and k-winner clusters, the cluster gain β is determined from Eqs. (35) and (36) of I to be

$$\beta=\begin{cases}\dfrac{Q\gamma}{2(Q-1)} & (\text{winner-take-all}) \qquad (29)\\[2ex] \dfrac{Q\gamma}{4k(Q-k)} & (k\text{-winner})\ . \qquad (30)\end{cases}$$

For large N, we find that the minimum eigenvalue of the interconnection matrix (8) is

$$\lambda_{\min}\cong-\frac{\alpha}{Q}\frac{k(Q-k)}{Q-1}\ . \qquad (31)$$

This result was found by numerically computing the eigenvalue spectra of computer-generated interconnection matrices; typical results for winner-take-all matrices appear in Fig. 9. The results of Fig. 9 were generated by constructing 20 matrices according to Eq. (8) for network sizes $N=50$, 75, 100, 150, and 200 for three sets of Q and α. For large N, the eigenvalue spectra are similar to that of a standard Hebb-rule matrix [47–49], with αN positive eigenvalues forming a continuous distribution, N degenerate eigenvalues equal to zero, and $N(Q-1-\alpha)$ negative eigenvalues grouped about the value $-\alpha/Q$. Combining Eq. (39) of I and Eqs. (29)–(31) yields the following expressions for the oscillation region boundary:

$$\hat{\gamma}=\begin{cases}\dfrac{1}{\alpha}\dfrac{2(Q-1)}{Q} & (\text{winner-take-all}) \qquad (32)\\[2ex] \dfrac{1}{\alpha}\dfrac{4k(Q-k)(Q-1)}{Q^2} & (k\text{-winner})\ . \qquad (33)\end{cases}$$

The storage capacity boundary terminates at the oscillation region boundary because $L(t)$ is not a Liapunov function in the oscillation region.

The winner-take-all phase diagrams, Figs. 7(a)–7(c), are similar but not identical to phase diagrams for stochastic associative memory networks of Potts spins at

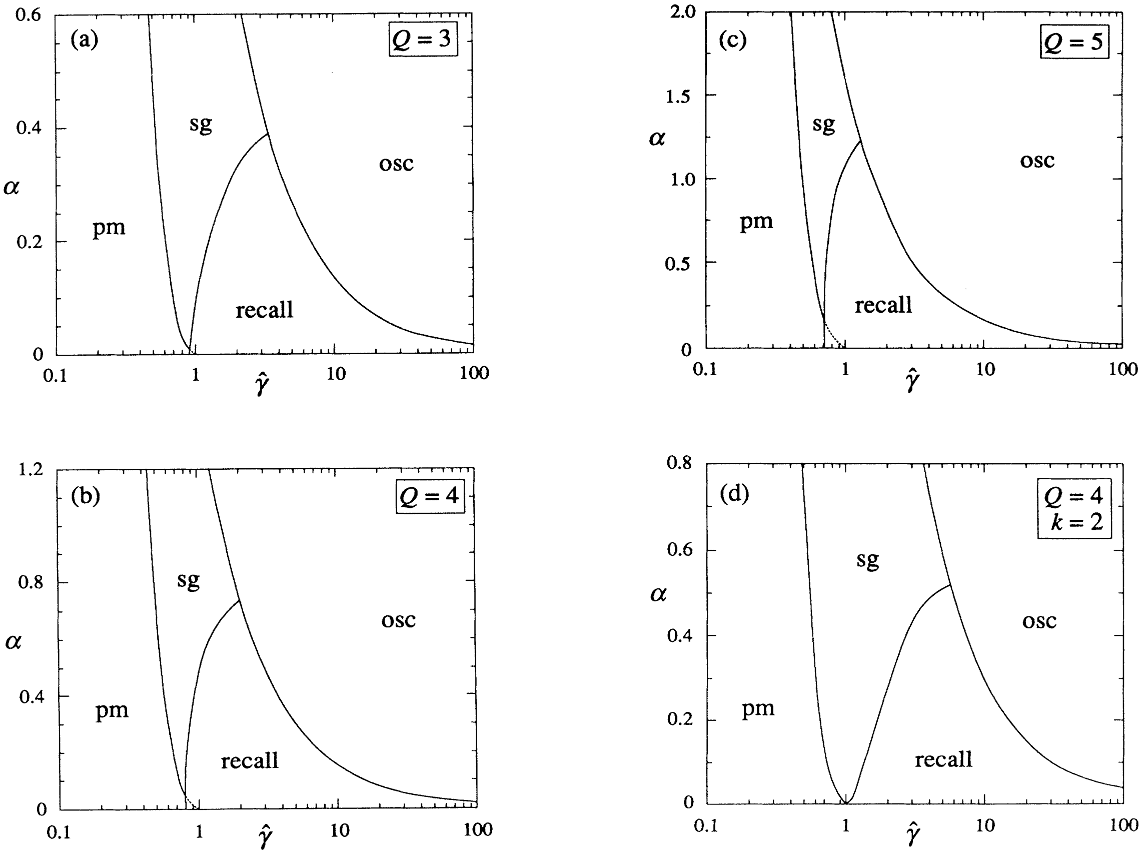


FIG. 8. Phase diagrams for extensively loaded associative memories with discrete-time parallel updating, showing different attractor types as function of storage fraction α and gain parameter $\hat{\gamma}$. First three diagrams are for winner-take-all networks with (a) $Q=3$, (b) $Q=4$, and (c) $Q=5$ neurons per cluster; fourth diagram (d) is for k-winner networks with $Q=4$ neurons and $k=2$ winners per cluster. Labels pm, sg, recall, and osc denote paramagnetic, spin-glass, recall, and oscillatory regions. Dashed curves within recall region in (a)–(c) indicate boundary between coexistence of recall attractors with paramagnetic attractor (low gain) and with spin-glass attractors (high gain). Outside oscillatory region, diagrams are identical to those for continuous-time updating in Fig. 7.

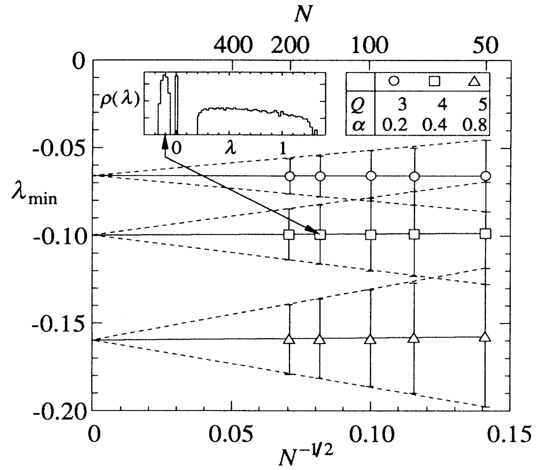


FIG. 9. Mean (markers) and standard deviation (error bars) of negative eigenvalues of computer-generated winner-take-all associative memory interconnection matrices for various values of Q and α, showing that distribution of negative eigenvalues approaches a δ-function peak at $\lambda_{\rm min}=-\alpha/Q$ as $N\rightarrow\infty$. Inset: Histogram of eigenvalue distribution for 20 matrices with $N=150$, $Q=4$, $k=1$, and $\alpha=0.4$. Vertical axis of histogram is logarithmic.

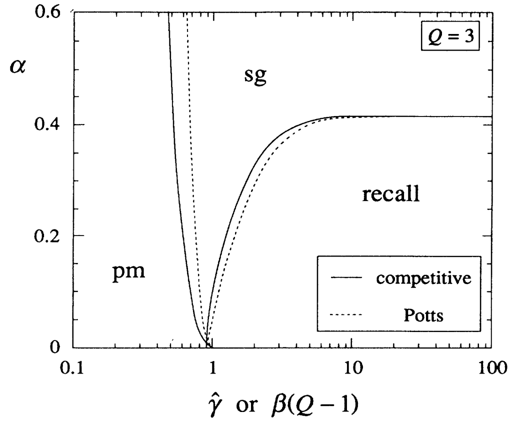


FIG. 10. Comparison of phase diagrams for winner-take-all competitive networks (solid curves) and Potts networks (dashed curves) for $Q=3$. Horizontal axis is gain parameter $\hat{\gamma}$ for competitive networks and $\beta(Q-1)$ for Potts networks, where β is inverse temperature. Data for Potts networks were supplied by authors of Refs. [38] and [39].

finite temperature [38,39]. Phase boundaries for the two network types are identical in the limits of infinite gain and of finite memory loading. The storage capacity values in the infinite-gain limit—which are $\alpha_c=0.414$, 0.828, and 1.375 for $Q=3$, 4, and 5—are the same as those reported in Ref. [33] for Potts networks. The values of $\hat{\gamma}$ at which recall attractors appear in the finite-loading limit—which are $\hat{\gamma}^*=0.915$, 0.805, and 0.713 for $Q=3$, 4, and 5—were reported in Refs. [34], [36], and [42] for Potts networks.

Aside from these limits, phase diagrams for analog competitive networks and Potts networks are *not* equivalent, due to differences between deterministic, analog dynamics and stochastic, discrete-state dynamics [23,24,26,27,50]. The mean-field treatment of finite-temperature Potts systems [51,52] yields a reaction field term [53] that subtracts each spin's influence from its own local field, whereas no reaction field appears in the neuron inputs (3) of competitive networks. This difference is illustrated in Fig. 10, which compares phase diagrams of winner-take-all competitive networks and Potts networks for the case $Q=3$. The phase boundaries for Potts networks are from Refs. [38] and [39] and are plotted as functions of $(Q-1)\beta$, where β is the inverse temperature. The paramagnetic–spin-glass transition and the spin-glass–recall transition both lie further to the right of the diagram for stochastic networks as compared to analog networks. An intuitive explanation is that the reaction field acts as an effective noise source in the stochastic Potts network, decreasing the temperature at which transitions occur.

VI. STORAGE CAPACITY IN INFINITE-GAIN LIMIT

For a given network configuration, the maximum value of the storage capacity α_c is usually achieved in the limit of infinite neuron gain, aside from small reentrant effects [39,54]. In this limit, the mean-field equations (21)–(23) simplify considerably. All but one dimension of the integrals can be done analytically, and the three equations can be reduced to one. These simplifications arise because outputs of high-gain neurons take on one of only two values, which are $1/k$ or $-1/(Q-k)$, respectively, for winning or losing neurons.

Infinite-gain storage capacities $\alpha_c(Q,k)$ are shown in Fig. 11 and Table I for $2\leq Q\leq 15$ and $1\leq k\leq 7$. The results hold also for zero-temperature Potts networks generalized to k-winner behavior, since the reaction field in these networks vanishes when $T=0$. The capacities obey

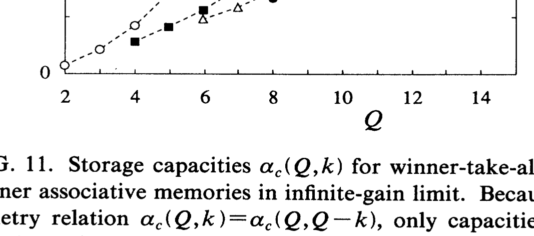


FIG. 11. Storage capacities $\alpha_c(Q,k)$ for winner-take-all and k-winner associative memories in infinite-gain limit. Because of symmetry relation $\alpha_c(Q,k)=\alpha_c(Q,Q-k)$, only capacities for $Q/k\geq 2$ are shown. Capacities also appear in Table I.

TABLE I. Storage capacities $\alpha_c(Q,k)$ of winner-take-all and k-winner associative memories in the limit of infinite neuron gain. Because of the symmetry relation $\alpha_c(Q,k) = \alpha_c(Q,Q-k)$, only capacities for $Q/k \geq 2$ are shown. Capacities are plotted in Fig. 11.

Q \ k	1	2	3	4	5	6	7
2	0.138						
3	0.414						
4	0.828	0.560					
5	1.375	0.799					
6	2.053	1.104	0.946				
7	2.859	1.468	1.159				
8	3.790	1.891	1.418	1.305			
9	4.844	2.371	1.718	1.501			
10	6.019	2.908	2.055	1.734	1.646		
11	7.312	3.502	2.430	1.998	1.831		
12	8.722	4.151	2.840	2.292	2.047	1.976	
13	10.25	4.856	3.287	2.612	2.288	2.153	
14	11.89	5.617	3.770	2.960	2.554	2.356	2.296
15	13.64	6.432	4.287	3.334	2.841	2.581	2.467

the symmetry relation $\alpha_c(Q,k)=\alpha_c(Q,Q-k)$, which is not outwardly apparent in the infinite-gain mean-field equations below. Some of the results for $k=1$ have been reported previously for Potts networks [33,38,39,42].

Figure 11 and Table I are calculated from the infinite-gain mean-field equations

$$m=-\frac{k}{Q-k}+\frac{Q}{Q-k}I_m\ , \tag{34}$$

$$C=\left[\frac{2Q}{\alpha rk(Q-k)(Q-1)}\right]^{1/2} I_C\ , \tag{35}$$

$$q=\frac{1}{Q-1}\ , \tag{36}$$

where

$$I_m=\int_{-\infty}^{\infty}\frac{dz}{\sqrt{\pi}}\exp(-z^2) \times\sum_{m=0}^{k-1}\sum_{n=0}^{Q-k+m} K_{mn}\{\tfrac{1}{2}[1+\mathrm{erf}(z)]\}^{Q-k+m-n} \times\{\tfrac{1}{2}[1+\mathrm{erf}(z+y)]\}^n \tag{37}$$

and

$$I_C=\int_{-\infty}^{\infty}\frac{dz}{\sqrt{\pi}}z\exp(-z^2)\sum_{m=0}^{k-1}\sum_{n=0}^{Q-k+m}(QK_{mn}\{\tfrac{1}{2}[1+\mathrm{erf}(z)]\}^{Q-k+m-n}\{\tfrac{1}{2}[1+\mathrm{erf}(z+y)]\}^n +(Q-k)L_{mn}\{\tfrac{1}{2}[1+\mathrm{erf}(z)]\}^n\{\tfrac{1}{2}[1+\mathrm{erf}(z-y)]\}^{Q-k+m-n})\ . \tag{38}$$

The coefficients K_{mn} and L_{mn} are

$$K_{mn}=\begin{bmatrix}k-1\\Q-k+m-n\end{bmatrix}\begin{bmatrix}Q-k\\n\end{bmatrix}(-1)^m \times\frac{1}{m!}\prod_{p=1}^{m}(Q-k+p-1)\ , \tag{39}$$

$$L_{mn}=\begin{bmatrix}k\\Q-k+m-n\end{bmatrix}\begin{bmatrix}Q-k-1\\n\end{bmatrix}(-1)^m \times\frac{1}{m!}\prod_{p=1}^{m}(Q-k+p-1)\ , \tag{40}$$

and the quantity y is defined as

$$y\equiv m\left[\frac{Q(Q-1)}{2\alpha rk(Q-k)}\right]^{1/2}\ . \tag{41}$$

Equations (34)–(36) reduce to a single equation for y:

$$y=\frac{Q-1}{Q-k}(QI_m-k)[2I_C+\sqrt{2\alpha k(Q-k)/Q}\ ]^{-1}\ . \tag{42}$$

The infinite-gain storage capacities α_c of Fig. 11 and Table I indicate the value of α at which the nonzero solution of Eq. (42) disappears. Because the replica-symmetric assumption breaks down for very large neuron gain, the storage capacities derived from Eq. (42) are not exact. By analogy with known results for Ising and Potts associative memories [38,39,44,51], we expect the exact storage capacities to be slightly higher than those appearing in Fig. 11 and Table I.

VII. NUMERICAL VERIFICATION OF PHASE DIAGRAMS

To test the phase diagrams derived in Sec. V, we have investigated the attractors of computer-generated competitive associative memories with discrete-time, parallel updating. Results are shown in Fig. 12. Each panel of Fig. 12 tests one of the phase diagrams in Fig. 8 along a horizontal line. The panels show, as a function of the gain parameter $\hat{\gamma}$, the fraction of randomly generated initial conditions that flow to each of the four possible types of attractors—the paramagnetic attractor, a recall attractor, a spin-glass attractor, or an oscillatory attractor. The first three panels in Fig. 12 are for three different winner-take-all network configurations: Fig. 12(a) is for networks with $N=100$ clusters, $Q=3$ neurons per cluster, and storage fraction $\alpha=0.2$; Fig. 12(b) is for networks with $N=75$, $Q=4$, and $\alpha=0.4$; and Fig. 12(c) is for networks with $N=60$, $Q=5$, and $\alpha=0.8$. The last panel, Fig. 12(d), is for k-winner networks with $N=75$ clusters, $Q=4$ neurons per cluster, $k=2$ winning neurons per cluster, and storage fraction $\alpha=0.2$.

The data in Fig. 12 were generated by starting from random initial conditions

$$x_{ia}(0)=\frac{1}{k(Q-k)}(Q\delta_{ab_i}-k)\ , \tag{43}$$

where each b_i, $i=1,\ldots,N$, is a set of k different integers chosen randomly and without bias from the set $\{1,\ldots,Q\}$. A total of 500 initial conditions were used for each panel, 20 from each of 25 different interconnec-

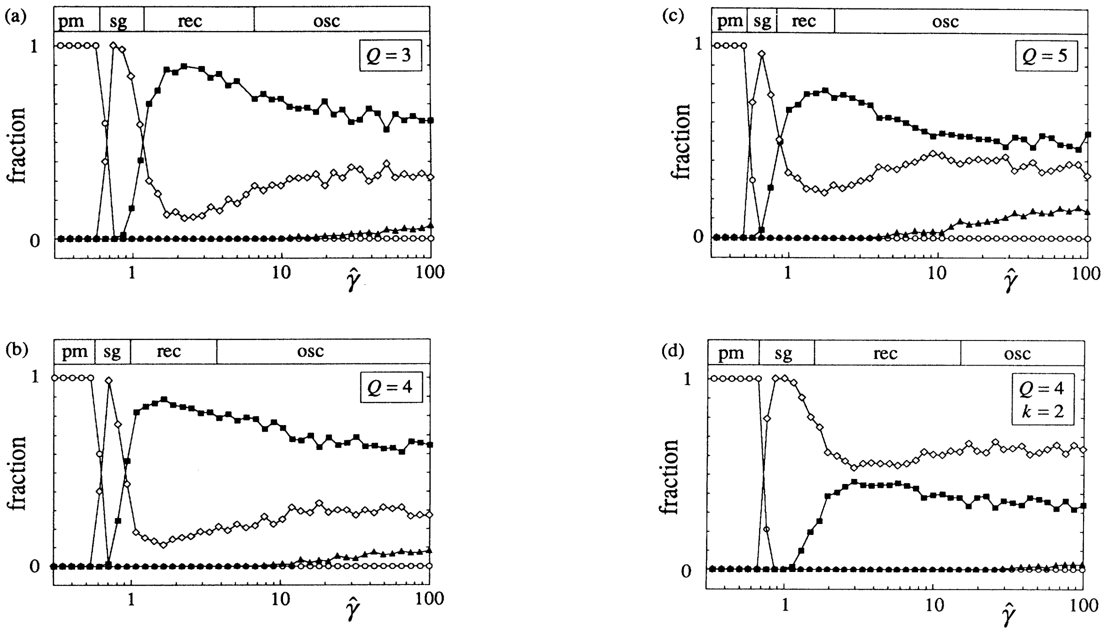


FIG. 12. Fraction of randomly generated initial conditions flowing to paramagnetic (circles), spin-glass (diamonds), recall (squares), and oscillatory (triangles) attractors in small, computer-generated competitive networks with discrete-time parallel updating. First three panels are for winner-take-all networks with (a) $N=100$, $Q=3$, and $\alpha=0.2$; (b) $N=75$, $Q=4$, and $\alpha=0.4$; and (c) $N=60$, $Q=5$, and $\alpha=0.8$. Fourth panel (d) is for k-winner networks with $N=75$, $Q=4$, $k=2$, and $\alpha=0.2$. Each panel tests one phase diagram of Fig. 8 along a horizontal line; at top of each panel, boxes labeled pm, sg, recall, and osc indicate corresponding phase-diagram regions. The data strongly support the phase diagrams.

tion matrices constructed according to Eq. (8). The updated equations (2) were then iterated until convergence to a fixed-point or period-two attractor. The fixed points were classified into three categories: recall attractors, for which one of the threshold overlaps

$$m_\mu^{\mathrm{thr}} \equiv \frac{1}{Nk} \sum_{i=1}^{N} \mathrm{sgn}(x_{i\xi_i^\mu}) \ , \tag{44}$$

is greater than 0.9; paramagnetic attractors, for which $x_{ia}=0$ for all i and a; and spin-glass attractors, which are all other fixed-point attractors.

At the top of each panel are four boxes containing the names of the different attractor types—recall, pm, sg, and osc—that can occur in discrete-time networks. These boxes indicate the location on the appropriate phase diagram in Fig. 8 of the four phase regions for the particular value of α used in that panel. The paramagnetic–spin-glass transitions and the spin-glass–recall transitions occur at values of $\hat{\gamma}$ predicted by the phase diagrams; the lack of sharpness in these transitions is the result of finite-size effects. Oscillatory attractors appear at values of $\hat{\gamma}$ somewhat higher than that given by the stability criterion, which is, however, a worst-case result. We discuss the delayed appearance of oscillatory attractors in competitive networks in paper I. One notable feature of Fig. 12 is that recall ability decreases as $\hat{\gamma}$ increases within the recall region. Improved performance at lower gain has been observed in a variety of analog systems [18,19,23,55,56] and has been investigated analytically in standard analog associative memories [24,25].

VIII. SUMMARY

We have combined techniques of nonlinear dynamics and the statistical mechanics of disordered systems to analyze associative memory in analog neural networks with localized competitive interactions. Our results, summarized in the phase diagrams of Sec. V, indicate that these networks work reliably as associative memories over a large range of storage fraction and transfer-function gain. The results are relevant to networks of resistively coupled nonlinear amplifiers that either evolve in continuous time or are clocked externally.

ACKNOWLEDGMENTS

We thank D. Bollé, A. Herz, J. Huyghebaert, R. Kühn, C. Marcus, and J. van Mourik for valuable discussions, and J. Huyghebaert for providing data appearing in Fig. 10. We also thank the Tinkham group at Harvard University for use of their computer. One of us (F.R.W.) acknowledges support from the U. S. Army Research Office. Research was supported in part by Contracts Nos. N00014-89-J-1023 and F49620-92-J-0466.

APPENDIX A

In this appendix, we derive the mean-field equations (12)–(14) for finite memory loading. Assuming all clusters to have the same Q transfer functions $F_a(z)$, $a=1,\ldots,Q$, and using Eq. (8) for the interconnection matrix, we write the free energy per neuron (9), averaged over all possible realizations of the patterns ξ_i^μ, as

$$f=-\frac{1}{\tilde{\beta}N}\left\langle \ln\int\prod_{i,a}[d\rho(x_{ia})]\exp\tilde{\beta}\left[\frac{1}{2N}\sum_{\mu}\left[\sum_{i}x_{i\xi_i^{\mu}}\right]^2-\frac{1}{2N}\sum_{i,\mu}\left[x_{i\xi_i^{\mu}}\right]^2-\sum_{i,a}G_a(x_{ia})\right]\right\rangle_{\xi} . \tag{A1}$$

The brackets $\langle\ \rangle_{\xi}$ indicate pattern averaging. The squared sum in the exponent is made linear in the neuron outputs using a Gaussian identity, which introduces the p overlaps m_{μ}, $\mu=1,\ldots,p$ [7,9]. In the limit $N\rightarrow\infty$, the second term in the exponent vanishes, the integrals over the overlaps can be done by saddle-point integration, and sums over the cluster index i are self-averaging. The resulting free energy is

$$f=\tfrac{1}{2}\sum_{\mu}m_{\mu}^2-\frac{1}{\tilde{\beta}}\left\langle \ln\int\prod_{a}[d\rho(x_a)]\exp\tilde{\beta}\left[\sum_{\mu}m_{\mu}x_{\xi^{\mu}}-\sum_{a}G_a(x_a)\right]\right\rangle_{\xi} , \tag{A2}$$

where the overlaps satisfy the saddle-point equations

$$m_{\mu}=\langle\langle x_{\xi^{\mu}}\rangle_x\rangle_{\xi} . \tag{A3}$$

The brackets $\langle\ \rangle_x$ in (A3) indicate an average over the integrand appearing in the free energy:

$$\langle\ \rangle_x\rightarrow\frac{\int\prod_{a}[d\rho(x_a)]I(\ )}{\int\prod_{a}[d\rho(x_a)]I} , \tag{A4}$$

where

$$I\equiv\exp\tilde{\beta}\left[\sum_{\mu}m_{\mu}x_{\xi^{\mu}}-\sum_{a}G_a(x_a)\right] . \tag{A5}$$

As discussed in Sec. III, we are interested only in the $\tilde{\beta}\rightarrow\infty$ limit of the free energy. In this limit, the integrals over the neuron outputs x_a can be done by saddle-point integration. The saddle-point equations are determined by maximizing I with respect to the x_a, subject to the competitive constraint $\sum_a x_a=0$. Using a Lagrange multiplier B to enforce this constraint leads to the following saddle-point equations:

$$x_a=\left\langle F_a\left[\sum_{\mu}m_{\mu}\delta_{a,\xi^{\mu}}+B\right]\right\rangle_{\xi} , \quad a=1,\ldots,Q . \tag{A6}$$

Inserting (A6) into the free energy (A2) and the overlaps (A3) leads to Eqs. (12) and (13).

APPENDIX B

The stability of the overlap solutions in finitely loaded networks is determined by the eigenvalue spectrum of the matrix $\partial^2 f/\partial m_{\rho}\partial m_{\sigma}$, where f is the free energy (12). A solution is stable if all eigenvalues of this matrix are positive. When all neurons have the same transfer function $F(z)$, the matrix is

$$\frac{\partial^2 f}{\partial m_{\rho}\partial m_{\sigma}}=\left\langle \delta_{\rho,\sigma}-\delta_{\xi^{\rho},\xi^{\sigma}}F'(h_{\xi^{\rho}}+B)+\frac{F'(h_{\xi^{\rho}}+B)F'(h_{\xi^{\sigma}}+B)}{\sum_{a}F'(h_a+B)}\right\rangle_{\xi} , \tag{B1}$$

where $h_a\equiv\sum_{\mu}m_{\mu}\delta_{a,\xi^{\mu}}$ and $F'(z)$ is the transfer-function derivative. In the case $m_{\mu}=m\,\delta_{\mu,1}$ of a single successfully recalled pattern, the matrix is diagonal, with the following two eigenvalues:

$$\lambda_1=1-\frac{1}{S}k(Q-k)F'(m+B)F'(B) , \tag{B2}$$

$$\lambda_2=1-\frac{1}{QS}[k(k-1)F'(m+B)^2+(Q-k)(Q-k-1)F'(B)^2+2k(Q-k)F'(m+B)F'(B)] , \tag{B3}$$

where

$$S\equiv kF'(m+B)+(Q-k)F'(B) . \tag{B4}$$

The eigenvalue λ_1 indicates stability of a solution with respect to perturbations of the nonzero overlap m_1, while λ_2 indicates stability with respect to perturbations of the other $p-1$ overlaps m_{μ}, $\mu>1$. Note that the eigenvalues λ_1 and λ_2, unlike the mean-field equation (20) for k-winner networks, are not invariant to replacing Q and k by nQ and nk for positive integers n. Thus the stability of solutions of Eq. (20) depends on Q, even though the solutions themselves depend only on the ratio Q/k.

APPENDIX C

In this appendix, we sketch the derivation of the mean-field equations (21)–(23) for extensive memory loading. The derivation follows the standard replica approach; we refer the reader to Refs. [7], [9], and [27] for details. In the replica approach, the identity

$$\ln z=\lim_{n\rightarrow 0}(z^n-1) \tag{C1}$$

is used to write the free energy (9) per neuron, averaged over all possible realizations of the patterns, as

$$f=\lim_{n\to 0}\left[-\frac{1}{\tilde\beta nN}\left\{\left\langle\int\prod_{i,a,\sigma}[d\rho(x_{ia}^{\sigma})]\exp\tilde\beta\left[\frac{1}{2N}\sum_{\mu,\sigma}\left[\sum_{i}x_{i\xi_i^{\mu}}^{\sigma}\right]^2-\frac{1}{2N}\sum_{i,\mu,\sigma}\left[x_{i\xi_i^{\mu}}^{\sigma}\right]^2-\sum_{i,a,\sigma}G_a(x_{ia}^{\sigma})\right]\right\rangle_{\xi}-1\right\}\right], \tag{C2}$$

where the brackets $\langle\ \rangle_{\xi}$ indicate pattern averaging. The index σ, which labels the replicas, runs from 1 to n. To make the squared sum in (C2) linear in the neuron outputs, the p pattern overlaps m_{μ}^{σ}, $\mu=1,\ldots,p$, are introduced using a Gaussian identity. The patterns are separated into a finite number s of "condensed" patterns with nonzero overlaps and an infinite number $p-s$ of "uncondensed" patterns with vanishing overlaps. The average over uncondensed patterns is performed assuming that each pattern ξ_i^{μ} occurs with equal probability $k!(Q-k)!/Q!$, and the overlaps corresponding to these patterns are integrated out. After using self-averaging to remove the site index i and taking the $N\to\infty$ limit, the free energy is

$$\begin{aligned}f=\lim_{n\to 0}\Bigg[&\frac{1}{n}\exp\left\{\tfrac{1}{2}\sum_{\nu,\sigma}(m_{\nu}^{\sigma})^2+\frac{\alpha}{2\tilde\beta}\operatorname{Tr}\ln(\mathbf{I}-\tilde\beta\mathbf{q}_0-\tilde\beta\mathbf{q})+\frac{\alpha\tilde\beta}{2}\left[\sum_{\sigma}q_0^{\sigma}r_0^{\sigma}+\sum_{\substack{\sigma,\sigma'\\(\sigma\neq\sigma')}}q^{\sigma\sigma'}r^{\sigma\sigma'}\right]\right\}\\&-\frac{1}{\tilde\beta}\Bigg\langle\ln\int\prod_{a,\sigma}[d\rho(x_a^{\sigma})]\exp\Bigg[\tilde\beta\sum_{\nu,\sigma}m_{\nu}^{\sigma}x_{\xi^{\nu}}^{\sigma}-\tilde\beta\sum_{a,\sigma}G_a(x_a^{\sigma})\\&+\frac{\alpha\tilde\beta}{2Q}\frac{k(Q-k)}{Q-1}\left\{\sum_{a,\sigma}(\tilde\beta r_0^{\sigma}-1)(x_a^{\sigma})^2+\tilde\beta\sum_{\substack{a,\sigma,\sigma'\\(\sigma\neq\sigma')}}r^{\sigma\sigma'}x_a^{\sigma}x_a^{\sigma'}\right\}\Bigg]\Bigg\rangle_{\xi}\Bigg]. \end{aligned}\tag{C3}$$

The quantities m_{ν}^{σ}, q_0^{σ}, r_0^{σ}, $q^{\sigma\sigma'}$, and $r^{\sigma\sigma'}$ are order parameters. The matrices $\mathbf{I}$, $\mathbf{q}_0$, and $\mathbf{q}_1$ in Eq. (C3) are $n\times n$; $\mathbf{I}$ is the identity matrix, $\mathbf{q}_0$ is a diagonal matrix with diagonal elements equal to q_0^{σ}, and $\mathbf{q}$ is a symmetric matrix with diagonal elements equal to 0 and off-diagonal elements equal to $q^{\sigma\sigma'}$. The pattern average $\langle\ \rangle_{\xi}$ is now over the s condensed patterns ξ^{ν}, $\nu=1,\ldots,s$.

We now assume replica symmetry, meaning that the values of the order parameters in Eq. (C3) are independent of replica index σ. Physically, replica symmetry means that there is only one fixed point in the vicinity of a stored memory. By analogy with Ising associative memories, for which replica-symmetry-breaking effects are small [44,45], we believe that the replica-symmetric solution is correct over a wide range of neuron gain but breaks down for very high gain.

Applying replica symmetry and taking the limit $n\to 0$ yields

$$\begin{aligned}f=&\tfrac{1}{2}\sum_{\nu}m_{\nu}^2+\tfrac{1}{2}\alpha\left\{\frac{1}{\tilde\beta}\ln[1-\tilde\beta(q_0-q)]-\frac{q}{1-\tilde\beta(q_0-q)}+q_0\tilde r+\tilde\beta(q_0-q)r\right\}\\&-\frac{1}{\tilde\beta}\left\langle\ln\int\prod_{a}[d\rho(x_a)]\exp\tilde\beta\left[\sum_{\nu}m_{\nu}x_{\xi^{\nu}}-\sum_{a}G_a(x_a)+\frac{\alpha}{2Q}\frac{k(Q-k)}{Q-1}(\tilde r-1)\sum_{a}x_a^2+\left[\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}\right]^{1/2}\sum_{a}z_ax_a\right]\right\rangle,\end{aligned}\tag{C4}$$

where $\tilde r\equiv\tilde\beta(r_0-r)$. The brackets $\langle\ \rangle_{z,\xi}$ stand for an average over both the s condensed patterns ξ^{ν} and the Q continuous variables z_a using a Gaussian distribution:

$$\langle\ \rangle_{z,\xi}\to\left\langle\int\prod_{a}\left[\frac{dz_a}{\sqrt{2\pi}}\right]\exp\left[-\tfrac{1}{2}\sum_{a}z_a^2\right](\ )\right\rangle_{\xi}. \tag{C5}$$

The saddle-point equations for m_{ν}, q_0, q, r, and $\tilde r$ are calculated by setting the partial derivatives of f with respect to these variables equal to zero:

$$m_{\nu}=\langle\langle x_{\xi^{\nu}}\rangle_x\rangle_{z,\xi}, \tag{C6}$$

$$\tilde\beta(q_0-q)=\left[\frac{1}{\alpha rQ}\frac{k(Q-k)}{Q-1}\right]^{1/2}\left\langle\left\langle\sum_{a}z_ax_a\right\rangle_x\right\rangle_{z,\xi} \tag{C7}$$

$$q_0=\frac{1}{Q}\frac{k(Q-k)}{Q-1}\left\langle\left\langle\sum_{a}x_a^2\right\rangle_x\right\rangle_{z,\xi}, \tag{C8}$$

$$r=\frac{q}{[1-\tilde\beta(q_0-q)]^2}, \tag{C9}$$

$$\tilde r=\frac{1}{1-\tilde\beta(q_0-q)}, \tag{C10}$$

where

$$\langle\ \rangle_x\to\frac{\int\prod_{a}[d\rho(x_a)]I(\ )}{\int\prod_{a}[d\rho(x_a)]I} \tag{C11}$$

and I is the integrand appearing in double brackets in f:

$$I=\exp\left\{\tilde{\beta}\left[\sum_{\nu} m_{\nu}x_{\xi^{\nu}}-\sum_{a} G_a(x_a)+\frac{\alpha}{2Q}\frac{k(Q-k)}{Q-1}(\tilde{r}-1)\sum_{a} x_a^2+\left[\frac{\alpha r}{Q}\frac{k(Q-k)}{Q-1}\right]^{1/2}\sum_{a} z_a x_a\right]\right\} . \quad \text{(C12)}$$

At the saddle point, the free energy can be written as

$$f=\tfrac{1}{2}\sum_{\nu} m_{\nu}^2+\tfrac{1}{2}\alpha\left\{\frac{1}{\tilde{\beta}}\ln[1-\tilde{\beta}(q_0-q)]+(q_0-q)\tilde{r}+\tilde{\beta}(q_0-q)r\right\}-\frac{1}{\tilde{\beta}}\left\langle \ln\int\prod_{a}[d\rho(x_a)]I\right\rangle_{z,\xi} . \quad \text{(C13)}$$

We are interested only in the $\tilde{\beta}\to\infty$ limit of the saddle-point equations (C6)–(C10). In this limit, the integrals over the neuron outputs x_a can themselves be carried out by saddle-point integration. This entails finding the values $\hat{x}_a$ for which the argument of the exponent in Eq. (C12) is a maximum, subject to the competitive constraint $\sum_a\hat{x}_a=0$. Using a Lagrange multiplier B to enforce this constraint leads to the Q saddle-point equations for the $\hat{x}_a$ that appear as Eq. (24). These values of $\hat{x}_a$ are then inserted into (C6)–(C8) to yield the saddle-point equations (21)–(23). Note that, since the right-hand side of Eq. (C7) is finite as $\tilde{\beta}\to\infty$, q_0 must approach q in such a way that $\tilde{\beta}(q_0-q)\equiv C$ approaches a finite value.

APPENDIX D

In this appendix, we derive the boundary (27) between the paramagnetic and spin-glass regions when the transition between these regions is continuous. The procedure is to expand Eq. (24) to leading order in the quantities $\hat{x}_a$, with all overlaps m_ν equal to zero. Inserting the result into the saddle-point equations (22) and (23) allows the integrals in these equations to be done analytically.

Expanding Eq. (24) around $\hat{x}_a=0$ when all neurons have the same transfer function $F(z)$ leads to

$$\hat{x}_a=\Phi[\alpha(\tilde{r}-1)U\hat{x}_a+\alpha r V z_a+\delta B] , \quad \text{(D1)}$$

where $\Phi\equiv F'(F^{-1}(0))$ and

$$U\equiv\frac{1}{Q}\frac{k(Q-k)}{Q-1} , \quad V\equiv\left[\frac{1}{\alpha r Q}\frac{k(Q-k)}{Q-1}\right]^{1/2} . \quad \text{(D2)}$$

The quantity δB, which is first-order in $\hat{x}_a$ and z_a, is determined by the competitive constraint $\sum_a\hat{x}_a=0$:

$$\delta B=-\frac{1}{Q}\sum_{a}[\alpha(\tilde{r}-1)U\hat{x}_a+\alpha r V z_a] . \quad \text{(D3)}$$

Inserting (D3) into (D1) and solving for $\hat{x}_a$ yields

$$\hat{x}_a=\frac{\alpha r V\Phi}{1-\alpha(\tilde{r}-1)U\Phi}\sum_{b} M_{ab}z_b , \quad \text{(D4)}$$

where $M_{ab}\equiv\delta_{ab}-1/Q$. The expansion (D4) is then inserted into the saddle-point equations (22) and (23) for q and C:

$$\begin{aligned} q&=U\left[\frac{\alpha r V\Phi}{1-\alpha(\tilde{r}-1)U\Phi}\right]^2\times\int\prod_{a}\left[\frac{dz_a}{\sqrt{2\pi}}\right]\exp\left[-\tfrac{1}{2}\sum_{a} z_a^2\right]\sum_{a,b,c} M_{ab}M_{ac}z_b z_c\\ &=U\left[\frac{\alpha r V\Phi}{1-\alpha(\tilde{r}-1)U\Phi}\right]^2(Q-1) , \end{aligned} \quad \text{(D5)}$$

$$\begin{aligned} C&=V\left[\frac{\alpha r V\Phi}{1-\alpha(\tilde{r}-1)U\Phi}\right]\times\int\prod_{a}\left[\frac{dz_a}{\sqrt{2\pi}}\right]\exp\left[-\tfrac{1}{2}\sum_{a} z_a^2\right]\sum_{a,b} M_{ab}z_a z_b\\ &=V\left[\frac{\alpha r V\Phi}{1-\alpha(\tilde{r}-1)U\Phi}\right](Q-1) . \end{aligned} \quad \text{(D6)}$$

Inserting the expressions (26) for r and $\tilde{r}$ into Eqs. (D5) and (D6) produces two equations in the two quantities Φ and C; eliminating C yields the boundary curve of Eq. (27).

[1] J. A. Anderson, Kybernetik, **5**, 113 (1968).

[2] D. J. Willshaw, O. P. Buneman, and H. C. Longuet-Higgins, Nature **222**, 960 (1969).

[3] S.-I. Amari, IEEE Trans. Comput. **C-21**, 1197 (1972).

[4] T. Kohonen, IEEE Trans. Comput. **C-23**, 444 (1974).

[5] W. A. Little, Math. Biosci. **19**, 101 (1974).

[6] J. J. Hopfield, Proc. Natl. Acad. Sci. U.S.A. **79**, 2554 (1982).

[7] D. J. Amit, H. Gutfreund, and H. Sompolinsky, Phys. Rev. A **32**, 1007 (1985); Phys. Rev. Lett. **55**, 1530 (1985); Ann. Phys. (N.Y.) **173**, 30 (1987).

[8] E. J. Gardner, J. Phys. A **21**, 257 (1988).

[9] Recent reviews and collections of articles include J. Amit, *Modeling Brain Function: The World of Attractor Neural Networks* (Cambridge University Press, Cambridge, MA, 1989); *Statistical Mechanics of Neural Networks: Proceedings of the XIth Sitges Conference, Sitges, Barcelona, 1990,* edited by L. Garrido (Springer-Verlag, New York, 1990); *Models of Neural Networks,* edited by E. Domany, J. L. van Hemmen, and K. Schulten (Springer-Verlag, New York, 1991); Physica A **185**, 343 (1992).

[10] F. R. Waugh and R. M. Westervelt, preceding paper, Phys. Rev. E **47**, 4524 (1993).

[11] S. C. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Murthy, Neural Networks **3**, 277 (1990).

[12] J. S. Bridle and S. J. Cox, in *Advances in Neural Information Processing Systems 3,* edited by R. P. Lippmann, J. E. Moody, and D. S. Touretsky (Kaufmann, San Mateo, CA,

1991), p. 234.

[13] A. Lapedes, C. Barnes, C. Burks, R. Faber, and K. Sirotkin, in *Computers and DNA, SFI Studies in the Sciences of Complexity,* edited by G. Bell and T. Marr (Addison-Wesley, Reading, MA, 1989), Vol. 7, p. 157.

[14] M. C. O'Neill, Nucleic Acids Res. **19**, 313 (1991).

[15] H. Bohr, J. Bohr, S. Brunak, R. Cotterill, H. Fredholm, B. Lautrup, and S. Petersen, FEBS Lett. **261**, 43 (1990).

[16] J. Rubner and K. Schulten, Biol. Cyber. **62**, 193 (1990).

[17] M. A. Cohen and S. Grossberg, IEEE Trans. SMC **13**, 815 (1983).

[18] J. J. Hopfield, Proc. Natl. Acad. Sci. U.S.A. **81**, 3008 (1984).

[19] J. J. Hopfield and D. W. Tank, Science **233**, 625 (1986).

[20] R. M. Golden, J. Math. Psych. **30**, 73 (1986).

[21] C. M. Marcus and R. M. Westervelt, Phys. Rev. A **40**, 501 (1989).

[22] F. Fogelman Soulie, C. Mejia, E. Goles, and S. Martinez, Complex Syst. **3**, 269 (1989).

[23] C. M. Marcus, F. R. Waugh, and R. M. Westervelt, Phys. Rev. A **41**, 3355 (1990).

[24] F. R. Waugh, C. M. Marcus, and R. M. Westervelt, Phys. Rev. Lett. **64**, 1986 (1990); Phys. Rev. A **43**, 3131 (1991).

[25] T. Fukai and M. Shiino, Phys. Rev. A **42**, 7459 (1990).

[26] M. Shiino and T. Fukai, J. Phys. A **23**, L1009 (1990).

[27] R. Kühn, S. Bös, and J. L. van Hemmen, Phys. Rev. A **43**, 2084 (1991).

[28] M. Shiino and T. Fukai, J. Phys. A **25**, L375 (1992).

[29] E. Goles-Chacc, F. Fogelman-Soulie, and D. Pellegrin, Disc. Appl. Math. **12**, 261 (1985); E. Goles and G. Y. Vichniac, in *Neural Networks for Computing,* AIP Conf. Proc. No. 151, edited by J. S. Denker (American Institute of Physics, New York, 1986), p. 165.

[30] K. Fukushima, Biol. Cyber. **36**, 193 (1980); Neural Networks **1**, 199 (1988).

[31] D. E. Remelhart and D. Zipser, in *Parallel Distributed Processing, Vol. 1,* edited by J. A. Feldman, P. J. Hayes, and E. E. Rumelhart (MIT Press, Cambridge, MA, 1986), p. 151.

[32] J. A. Hertz, A. S. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Reading, MA, 1991), Chap. 9.

[33] I. Kanter, Phys. Rev. A **37**, 2739 (1988).

[34] D. Bollé and F. Mallezie, J. Phys. A **22**, 4409 (1989).

[35] J.-P. Nadal and A. Rau, J. Phys. I (Paris) **1**, 1109 (1991).

[36] D. Bollé, P. Dupont, and J. van Mourik, J. Phys. A **24**, 1065 (1991).

[37] D. Bollé, P. Dupont, and B. Vinck, J. Phys. A **25**, 2859 (1992).

[38] D. Bollé, P. Dupont, and J. Huyghebaert, Phys. Rev. A **45**, 4194 (1992).

[39] D. Bollé, P. Dupont, and J. Huyghebaert, Physica A **185**, 363 (1992).

[40] P. A. Ferrari, S. Martinez, and P. Picco, J. Stat. Phys. **66**, 1643 (1992).

[41] G. M. Shim, D. Kim, and M. Y. Choi, Phys. Rev. A **45**, 1238 (1992).

[42] H. Vogt and A. Zippelius, J. Phys. A **25**, 2209 (1992).

[43] J. F. Fontanari and R. Köberle, J. Phys. (Paris) **49**, 13 (1988).

[44] A. Crisanti, D. J. Amit, and H. Gutfreund, Europhys. Lett. **2**, 337 (1986).

[45] G. A. Kohring, J. Stat. Phys. **59**, 1077 (1990).

[46] C. M. Marcus and R. M. Westervelt, Phys. Rev. A **39**, 347 (1989).

[47] S. Geman, Ann. Prob. **8**, 252 (1980).

[48] A. Crisanti and H. Sompolinsky, Phys. Rev. A **36**, 4922 (1987).

[49] Y. Le Cun, I. Kanter, and S. A. Solla, Phys. Rev. Lett. **66**, 2396 (1991).

[50] H. Takayama and K. Nemoto, J. Phys.: Condens. Matter **2**, 1997 (1990).

[51] E. J. S. Lage and J. M. Nunes da Silva, J. Phys. C **17**, L593 (1984).

[52] D. J. Gross, I. Kanter, and H. Sompolinsky, Phys. Rev. Lett. **55**, 304 (1984).

[53] D. J. Thouless, P. W. Anderson, and R. G. Palmer, Philos. Mag. **35**, 593 (1977).

[54] J.-P. Naef and A. Canning, J. Phys. I (Paris) **2**, 247 (1992).

[55] C. Koch, J. Marroquin, and A. Yuille, Proc. Natl. Acad. Sci. U.S.A. **83**, 4263 (1986).

[56] R. Durbin and D. Willshaw, Nature **326**, 689 (1987).