Neural networks with many-neuron interactions

G.A. Kohring

Tome 51 N° 2 15 JANVIER 1990

LE JOURNAL DE PHYSIQUE

J. Phys. France **51** (1990) 145-155 15 JANVIER **1990**, PAGE 145

Classification
Physics Abstracts
87.30G — 75.10H — 64.60C

Neural networks with many-neuron interactions

G. A. Kohring (*)

Physikalisches Institut der Universität Bonn, Nussallee 12, D-5300 Bonn 1, F.R.G.

(*Reçu le 2 octobre 1989, accepté le 3 octobre 1989*)

Résumé. — Les propriétés statiques et dynamiques de réseaux neuronaux avec interactions entre plusieurs neurones sont étudiées analytiquement et numériquement. La capacité de stockage de ces réseaux est la même que celle des réseaux avec interactions entre paires de neurones, ce qui implique que ces réseaux ne stockent pas l'information de manière plus efficace. La taille des bassins d'attractions est calculée exactement à partir d'une solution de la dynamique du réseau totalement connecté. Ceci montre que les réseaux avec interactions entre plus de deux neurones sont plus efficaces pour la reconnaissance des *patterns* que les réseaux avec interactions entre paires de neurones.

Abstract. — The static and dynamical properties of neural networks having many-neuron interactions are studied analytically and numerically. The storage capacity of such networks is found to be unchanged from that of the more widely studied case of two-neuron interactions implying that these networks store information no more efficiently. The size of the basins of attraction in the many-neuron case is calculated exactly from a solution of the network dynamics at full connectivity and reveals that networks with many-neuron interactions are better at pattern discrimination than the simpler networks with only two-neuron interactions.

Introduction.

The static structure of the simplest single layer neural networks is described by N two-state neurons, S_i, and a real-valued, $N \times N$ coupling matrix, T_{ij}, which contains information about the architecture of the network as well as determines the stable states. The two-dimensional character of the coupling matrix follows from the assumption that the neurons have only two-neuron interactions. However, it was realized quite early [1] that this assumption is not tenable for biological networks. Biological networks have a propensity for many-neuron interactions, although the reasons for this are far from clear. Several authors [1, 2, 3] have

(*) *Present address :* HLRZ c/o KFA Jülich, D-5170 Jülich 1, F.R.G.

pointed out the increased storage capacity of networks using multibody interactions and Baldi and Venkatesh [3] have estimated that this storage capacity increases like $O(N^{\lambda})$ for networks having $(\lambda + 1)$-neuron interactions. For the case of two-neuron interactions it is known from the work of Venkatesh [4] and Gardner [5] that the maximum number of uncorrelated patterns, P, which can be stored without error, increases as $P = 2N$. If one defines the *storage capacity,* α, as :

$$\alpha = \lim_{N \to \infty} \frac{\text{total number of bits/neuron to be stored}}{\text{total number of synaptic couplings/neuron}} \tag{1.1}$$

then the maximum storage capacity for networks having two-neuron interactions is $\alpha = 2$. Hence, the important question to be asked about networks with many-neuron interactions is whether this storage capacity is increased, decreased or remains the same. In other words, do many-neuron connections store information more or less efficiently than two-neuron connections ? The results to be derived here will show that the maximum obtainable value of α is $\alpha = 2$ for all values of $\lambda \geqslant 1$.

In addition to their static properties, neural networks are also defined by their dynamical properties. The most common dynamics studied, for two-neuron interactions, is defined, at zero temperature (i.e. in the absence of noise) *via* the equation :

$$S_i(t+1) \equiv \text{sgn}\left(\sum_{j \neq i} J_{ij} S_j(t)\right) . \tag{1.2}$$

For strongly connected networks with only two-body interactions, the only dynamical property known analytically is the first step in this time evolution [6]. However, many other properties, such as the size of the basins of attraction are known from computer simulations [7-9, 15]. A natural generalization of the dynamics to the multi-body case is defined by :

$$S_i(t+1) \equiv \text{sgn}\left(\sum_{j_1, \ldots, j_\lambda \neq i} T_{ij_1, \ldots, j_\lambda} S_{j_1}(t) \ldots S_{j_\lambda}\right) . \tag{1.3}$$

It turns out that multi-body dynamics are somewhat easier to study than two-body dynamics because of a *decrease* in the number of macroscopic variables needed to describe a given state of the network. Because of this decrease an ansatz can be made regarding the dynamics which allows certain parameters of the time evolution to be calculated exactly. These parameters, as will be shown, are in excellent agreement with computer simulations.

The plan of the paper is then as follows. In the next section the many-neuron networks will be defined in more detail and the static properties will be calculated analytically *via* a generalization of the method first employed by Gardner [5]. Then the first time step will be calculated analytically and compared to previous results. A conjecture will made at this point regarding the analytic form of the exact solution to the dynamical evolution. Next, after presenting a learning rule suitable for all values of λ, numerical results will be discussed which support the validity of the foregoing analytical arguements.

Static properties.

The properties of a neural network will depend, of course, upon the network architecture. For many-neuron interacting networks a fully connected architecture has some symmetries not present in the two-neuron case. This is illustrated schematically in figure 1 for the simple case of three-neuron interactions with directed synapses. Obviously, T_{ijk} is invariant with

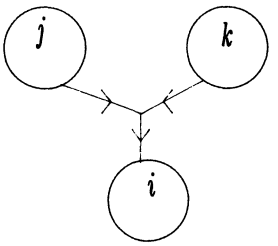

Fig. 1. — A schematic diagram of a three-neuron interaction.

respect to interchanges of the indices j and k, i.e., physically there exists only one three-neuron « synapse » labelled i-j-k. For a general, $(\lambda + 1)$-neuron interaction the synaptic matrix $T_{ij_1 \ldots j_\lambda}$ will be invariant under interchange of any of the j_1 to j_λ indicies. This should be reflected in the calculation of the local field by summing only over the independent elements of the synaptic matrix, i.e., the sum in equation (1.3) should be further restricted to run over all $j_1 < j_2 \cdots < j_\lambda$ in addition to the normal restriction of running over j_1, j_2, ..., $j_\lambda \neq i$.

With these extra symmetries, it is evident from equation (1.3) that the stable states of the system (denoted by ξ) will be defined as those states with local stability greater than zero :

$$\mathfrak{L}_i^\mu \equiv \frac{1}{\mathcal{N}} \sum_J T_{ij_1 \ldots j_\lambda} \xi_i^\mu \xi_{j_1}^\mu \ldots \xi_{j_\lambda}^\mu > 0 \tag{2.1}$$

for every neuron i. Where, $\mu \in \{1, \ldots, P\}$, $\mathcal{N}$ is defined as :

$$\mathcal{N} \equiv \sqrt{\binom{N-1}{\lambda}}, \tag{2.2}$$

which is simply the square root of the number of independent elements in the sum and $\sum_J$ denotes the restricted sum over only the independent elements of the synaptic matrix as described above.

In order to fix the scale for the couplings a suitable generalization of the two-neuron interaction case to the many-neuron interaction case is :

$$\sum_J T_{ij_1 \ldots j_\lambda}^2 = \mathcal{N}^2 . \tag{2.3}$$

For $\lambda = 1$, this is seen to reduce to the usual normalization [5] and as is the case there, it can be shown that this is not a real restriction on the network rather it is a *description* of the networks which emerge when constructed from some of the known learning rules (see Sect. 3). Experience with $\lambda = 1$ networks has also shown the need to adjust the stability condition in equation (2.1) from $\mathfrak{L}_i^\mu > 0$ to $\mathfrak{L}_i^\mu > \kappa$ (where κ is a positive constant which is taken for convience to be independent of μ and i), so as to produce stable states with non-negligible basins of attraction.

With the networks so defined, the calculation of the maximum storage capacity is done à la Gardner [5], i.e., the typical fractional volume of the coupling space which satisfies (2.1) and (2.3) at a given storage capacity is calculated and the maximum storage capacity is then

defined by the vanishing of this volume. The fractional volume is simply : $V_T = \prod_i V_i$, where V_i is given by :

$$V_i \equiv \frac{1}{Z_i} \int \mathfrak{D}T_i \prod_\mu \theta(\mathfrak{L}_i^\mu - \kappa) \tag{2.4}$$

with,

$$Z_i = \int \mathfrak{D}T_i \quad \text{and} \quad \mathfrak{D}T_i \equiv \prod_J \mathrm{d}T_{ij_1 \dots j_\lambda}\, \delta\left(\sum_J T^2_{ij_1 \dots j_\lambda} - \binom{N-1}{\lambda}\right).$$

The typical fractional volume is then to be found by averaging V_T over all realizations of the stored patterns. Since V_T factorizes over the index i, the assumption of self averaging with respect to i should be valid. Therefore, write, $\ln V_T = \sum_i \ln V_i$ and perform the quenched average on $\ln V_T$, i.e., on each subsystem $\ln V_i$. This average can be done *via* the replica method :

$$\langle\!\langle \ln V_i \rangle\!\rangle_\xi = \lim_{n \to 0} \frac{\langle\!\langle V_i^n \rangle\!\rangle_\xi - 1}{n}. \tag{2.5}$$

For the definition of V_i given above, the calculation of the quenched average reduces to evaluating the following expression :

$$\langle\!\langle V_i^n \rangle\!\rangle_\xi = \frac{1}{2^{NP} Z_i^n} \sum_\xi \int \prod_a \mathfrak{D}T_i^a \prod_{a\mu} \theta(\mathfrak{L}_i^{\mu a} - \kappa). \tag{2.6}$$

Using the standard integral representation of the θ function, this equation becomes :

$$\langle\!\langle V_i^n \rangle\!\rangle_\xi = \frac{1}{2^{NP} Z_i^n} \sum_\xi \int \prod_a \mathfrak{D}T_i^a \prod_\mu \mathrm{d}x_\mu^a \frac{y_\mu^a}{2\pi} \exp\left\{ i \sum_{a\mu} y_\mu^a (x_\mu^a - \mathfrak{L}_i^{\mu_a}) \right\} \tag{2.7}$$

where the integrals over the x-variables are restricted to the range $[\kappa, \infty]$ and the other integrals remain unrestricted. In order to calculate the average over the patterns, we rewrite the exponential as :

$$\frac{1}{2^{NP}} \sum_\xi \prod_{J\mu} \exp\left(-\frac{i}{\mathcal{N}} \sum_a y_\mu^a\, T^a_{ij_1 \dots j_\lambda}\, \xi_i^\mu\, \xi_{j_1}^\mu \dots \xi_{j_\lambda}^\mu \right) =$$

$$= \sum_\xi \prod_{J\mu} \cos\left(\frac{1}{\mathcal{N}} \sum_a y_\mu^a\, T^a_{ij_1 \dots j_\lambda} \right) \left[1 - i \xi_i^\mu\, \xi_{j_1}^\mu \dots \xi_{j_\lambda}^\mu \tan\left(\frac{1}{\mathcal{N}} \sum_a y_\mu^a\, T^a_{ij_1 \dots j_\lambda} \right) \right]. \tag{2.8}$$

The sum over the patterns acts only on the terms in the square brackets and when expanding these brackets out for $\lambda = 1$ they vanish identically. For $\lambda > 1$, all of those terms vanish in which ξ_k^μ does not occur an even number of times, hence, the lowest order contribution to this sum is :

$$\frac{1}{\mathcal{N}^{\frac{\lambda+1}{2}}} \sum_{j_1 < \dots < j_{\lambda(\lambda+1)/2}} T^a_{ij_1 \dots j_\lambda} \dots T^{a_\lambda}_{i \dots j_{\lambda(\lambda+1)/2}}. \tag{2.9}$$

Since the patterns are random and uncorrelated, *a priori* one expects that the $T_{ij_1 \dots j_\lambda}$ will also be uncorrelated. Hence, with on the order of $N^{\lambda(\lambda+1)/2}$ terms in the above sum and $\mathcal{N}^{(\lambda+1)/2} \sim N^{\lambda(\lambda+1)/2}$, these sums are of the order $N^{-\lambda(\lambda+1)/4}$. For large N these terms are negligible and one is left with :

$$\prod_{J\mu} \cos\left(\frac{1}{\mathcal{N}} \sum_a y_\mu^a \, T_{ij_1 \dots j_\lambda}^a\right) . \tag{2.10}$$

Therefore, to lowest order in $1/N$, the quenched average has the form :

$$\langle\!\langle V_i^n \rangle\!\rangle = \frac{1}{Z_i^n} \int \prod_a \mathfrak{D} T_i^a \prod_\mu \mathrm{d}x_\mu^a \frac{y_\mu^a}{2\pi} \times \exp\left\{ i \sum_{a\mu} y_\mu^a \, x_\mu^a - \frac{1}{2} \sum_\mu \sum_{ab} y_\mu^a \, y_\mu^b \left(\frac{1}{\mathcal{N}^2} \sum_J T_{ij_1 \dots j_\lambda}^a \, T_{ij_1 \dots j_\lambda}^b \right) \right\} . \tag{2.11}$$

If the variable, q^{ab}, is now introduced ;

$$q^{ab} \equiv \frac{1}{\mathcal{N}^2} \sum_J T_{ij_1 \dots j_\lambda}^a \, T_{ij_1 \dots j_\lambda}^b \qquad \forall a < b , \tag{2.12}$$

then equation (2.11) becomes identical to that found by Gardner for the case of two-neuron interactions. Hence, the rest of the analysis proceeds identically to her calculation and the interested reader is referred to her original paper for the details. The main result, obtained at the replica symmetric point, which can be shown to be stable, is that the storage capacity grows at saturation, i.e., the maximum allowed α for a given κ, as :

$$\alpha = \lim_{N \to \infty} \frac{P}{\binom{N-1}{\lambda}} = \left\{ \int_{-\kappa}^{\infty} \frac{\mathrm{d}t}{\sqrt{2\pi}} \mathrm{e}^{-t^2/2} (t + \kappa)^2 \right\}^{-1} . \tag{2.13}$$

When $\kappa = 0$ the patterns are minimally stable and the storage capacity has its maximum value of $\alpha = 2$. This proves that networks having many-neuron interactions store information no more efficiently than networks having only two-neuron interactions. They do store *much more* information than two-neuron connections, with the number of storable patterns growing as $N^\lambda/\lambda!$, however, they require of the same order of magnitude more connections to reach this higher level of storage.

Dynamic properties.

The dynamical behaviour of these models is at least as important as the storage properties for application purposes. In order to function as an associative memory, the stable states must be able to attract almost all states within a « reasonable » Hamming distance, i.e., the distance must not be so large that states having spurious correlations with the stored state are attracted, but it also must not be so small that only states which are almost identical to the stored state are attracted. The determination of this distance is a very difficult problem analytically. For strongly connected networks with two-neuron interactions, only the first step of the time evolution is known exactly [6], although the attraction basins have been studied by computer simulations [7-9].

To gain a simple understanding of dynamics for $\lambda > 1$, first compute the probability, $P(m(1)|m(0))$, that a state of the network at time, $t = 1$, has overlap, $m(1)$, $(m(t) \equiv (1/N) \sum_i \xi_i \, S_i(t))$ with some one of stored states, given that it had overlap $m(0)$ with the

same stored state at time $t = 0$ and random overlap with all of the other stable states. This probability is given by :

$$P(m(1)|m(0)) =$$

$$= \frac{\sum_{S(0)_i} \delta\left(m(1) - \frac{1}{N}\sum_i \operatorname{sgn}\left(\frac{1}{\mathcal{N}}\sum_J \xi_i\, T_{ij_1 \ldots J_\lambda}\, S_{j_1}(0) \ldots S_{j_\lambda}(0)\right)\right) \delta\left(m(0) - \frac{1}{N}\sum_j S_j(0)\,\xi_j\right)}{\sum_{S_i(0)} \delta\left(m(0) - \frac{1}{N}\sum_j S_j(0)\,\xi_j\right)} . \tag{3.1}$$

The evaluation of this probability proceeds along the lines of Kepler and Abbott's [6] work for the two-neuron interaction network and with the use of the same arguement as in section 2 for computing the sum over configurations, hence, omitting the details, the result is quite simply :

$$P(m(1)|m(0)) = \delta\left(m(1) - \frac{1}{N}\sum_i \operatorname{erf}\left(\frac{\mathfrak{L}_i\, m^\lambda(0)}{\sqrt{2[1 - m^{2\lambda}(0)]}}\right)\right), \tag{3.2}$$

where $\mathfrak{L}_i$ is defined in equation (2.1). If the distribution of the $\mathfrak{L}_i$'s is known, then (3.2) gives an explicit equation for $m(1)$. However, the only restriction on the $\mathfrak{L}_i$'s is : $\mathfrak{L}_i > \kappa \quad \forall i$, which is independent of λ ; hence, they should have the same distribution as for $\lambda = 1$ [10]. (It is very easy to check this *via* a replica calculation similar to that of Sect. 2.) Thus, with probability one, $m(1)$ becomes :

$$m(1) = \int_{-\infty}^{\infty} \mathrm{d}\mathfrak{L}\, \operatorname{erf}\left(\frac{\mathfrak{L} m^\lambda(0)}{\sqrt{2[1 - m^{2\lambda}(0)]}}\right) \times$$

$$\times \left[\frac{\mathrm{e}^{-\mathfrak{L}^2/2}}{\sqrt{2\pi}}\,\theta(\mathfrak{L} - \kappa) + \frac{1}{2}\operatorname{erfc}(-\kappa/\sqrt{2})\,\delta(\mathfrak{L} - \kappa)\right]. \tag{3.3}$$

For $\lambda = 1$, the calculation of $m(2)$ is very difficult because it depends not only upon $m(1)$ but also upon the parameter, $\rho(1) \equiv \sum_\mu [m^\mu(1)]^2$, which measures the projection of the state of the network onto the subspace spanned by the stored states. (At the first time step this parameter does not enter explicitly since the initial state of the network has been assumed to have macroscopic overlap with only one stored pattern and random overlap with all of the other stored patterns.) The calculation of the time evolution of $\rho(t)$ appears to be an intractable problem [11]. When $\lambda > 1$, however, a similar variable, $\rho_\lambda(t) \propto \sum_\mu [m^\mu(1)]^{2\lambda}$, does not play such a fundamental role because the complete configuration space is always spanned by the set of stored patterns (provided of course that $\alpha \neq 0$ and in the limit $N \to \infty$), so that ρ_λ is independent of time. The effect of this time independence can be seen by considering the basins of attraction around the stored states.

A convenient measure of the « size » of the basins of attraction around a given stored state can be computed from equation (3.3). This measure is often called the *radius of attraction* and it is denoted by R [15], with R being defined as : $R \equiv 1 - m_u$, and m_u is the unstable fixed point of (3.3) at a given value of κ (and as a consequence of Eq. (2.13) an equivalent value of α). R is plotted in figure 2 as a function of α for several values of λ. If $m(0) > m_u$, then the network will iterate toward the stable fixed point, $m = 1$. If on the other hand, $m(0) < m_u$ the network will iterate toward the stable fixed point $m = 0$. When $\lambda = 1$, it has

been noticed that the system may iterate in the first few time steps toward the fixed point $m = 1$ and then at latter times reverse direction and flow toward $m = 0$ [12]. Once, the system starts flowing toward $m = 0$ however, it has not been observed to reverse itself. (For the related case of feed-forward networks one can solve the dynamics exactly and prove rigorously that such an effect does indeed exist [13].) Thus, it can be conjectured, and is supported by the data (see next Sect.), that figure 2 gives the upper bound for the radius of attraction when $\lambda = 1$. The reversal of the network's direction of flow has previously been related to the time evolution of $\rho(t)$ [11]. Since, for $\lambda > 1$, ρ_λ should be independent of time, it can be conjectured that no such reversal takes place and that the fixed points of equation (3.3) determine the radius of attraction exactly. In the next section, numerical simulations will show this in fact to be the case.

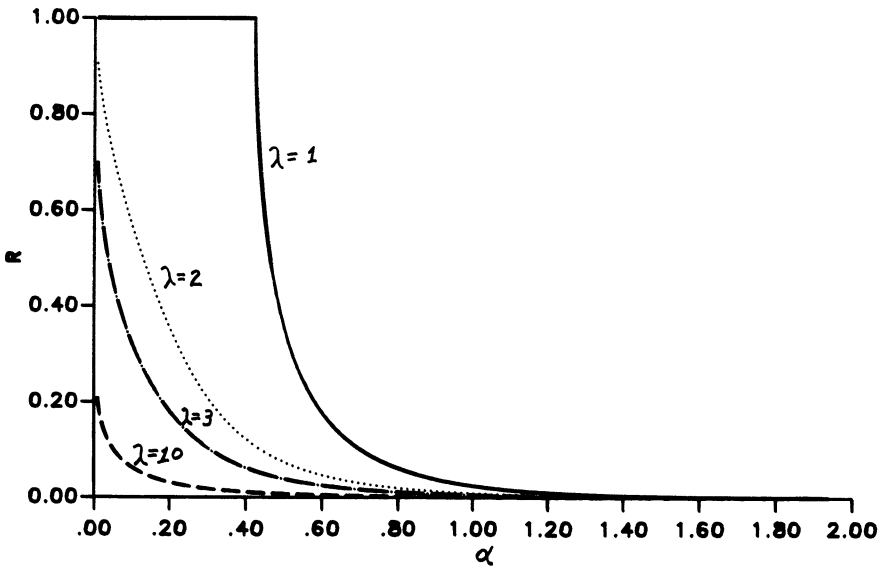

Fig. 2. — $R = 1 - m_c$, where m_c are the fixed points of equation (3.3), *vs.* α for several values of λ.

There is another qualitative difference between the flow diagram for $\lambda = 1$ and $\lambda > 1$, namely the nature of the transition to $R = 1$. For $\lambda = 1$, equation (3.3) predicts a second order transition to $R = 1$ at the value of κ defined by :

$$\frac{e^{-\kappa^2/2}}{\sqrt{2\pi}} + \frac{\kappa}{2}\,\mathrm{erfc}\left(-\frac{\kappa}{\sqrt{2}}\right) = \frac{\sqrt{2\pi}}{2}, \tag{3.4}$$

or $\alpha \approx 0.419$. While, for $\lambda > 1$, equation (3.3) predicts a first order jump to $R = 1$ at $\alpha = 0$. The size of this jump increases as λ increases and goes to 1 as $\lambda \to \infty$. In other words, as $\lambda \to \infty$, $R \to 0$ for all $\alpha > 0$ and $R = 1$ at $\alpha = 0$.

For the dynamical properties then, there are quantitative as well as qualitative differences between networks with only two-neuron interactions and networks with many-neuron interactions. In the following section these predicted differences will be validated *via* numerical simulations.

Numerical simulations.

A learning algorithm to generate networks satisfying the constraints of section two for many-neuron interacting networks is easily constructed from a generalization of the learning rules developed for two-neuron interacting networks [5, 8, 14]. Although any of the proposed learning rules can be generalized, the generalization here is based upon the learning rule developed by the Edinburgh group. Starting with the first pattern and moving sequentially through all the patterns (and in a sequential or parallel manner through the neurons), define an « error mask » as :

$$\varepsilon_i^\mu = \theta\left[\kappa \sqrt{\sum_J T^2_{ij_1 \dots j_\lambda}} - \sum_J T_{ij_1 \dots j_\lambda} \xi_i^\mu \xi_{j_1}^\mu \dots \xi_{j_\lambda}^\mu\right], \tag{4.1}$$

then update the coupling matrix as :

$$\Delta T_{ij_1 \dots j_\lambda} = \varepsilon_i^\mu \xi_i^\mu \xi_{j_1}^\mu \dots \xi_{j_\lambda}^\mu . \tag{4.2}$$

After all the patterns have been checked in this manner the procedure is repeated until the stability condition is satisfied for every pattern on every neuron. This learning algorithm can be shown to converge to a solution of the stability conditions provided such a solution exists. The proof makes no essential departures from that given in reference [5] for the case of two-neuron interaction and so will not be repeated here.

For finite size networks, one must compensate for the deviation of α_c from that given by equation (2.13). This is done by looking for the value of κ_N at which the learning time tends to infinity. κ_N will be somewhat less than the κ_c predicted by equation (2.13), but the finite size system will have properties which are very close to those of the infinite volume system [9].

As a test of the foregoing analytical calculations, $m(1)$ was measured as a function of $m(0)$ for $\alpha = 0.20$ and $\alpha = 0.4$ at $\lambda = 2$ and a network size of $N = 64$. As seen in figures 3a and 3b the results agree quite nicely with the analytic predictions. As $m(0)$ becomes small the interference from the other stored states increases (*a priori* one expects interference to set in around $m(0) \sim O(1/\sqrt{N})$) and the agreement with the theoretical calculations is predictably worse. Now, in the previous section it was conjectured that the radius of attraction could be calculated from equation (3.3) for $\lambda > 1$. To show that this is indeed the case a definition of R is needed which takes the finite size of the network into account.

A working definition of the radius of attraction R is ; $1 - m_c$, where m_c is the value of the overlap such that as $N \to \infty$ *almost all* of the states having $m > m_c$ will evolve toward $m = 1$. For finite size systems, R is reduced due to the $O(1/\sqrt{N})$ overlap between the stored states. Hence, a better definition of R is [15] : $R = \left\langle\!\!\left\langle \frac{1 - m_c}{1 - m_{av}} \right\rangle\!\!\right\rangle$, where m_{av} is the average overlap of the given state with all of the other stored states and the bracket indicates an average over all stored states and all starting configurations. (As $N \to \infty$, the m_c defined here tends to the m_u defined in the previous section and m_{av} tends toward zero.)

The results obtained when calculating R using this prescription are plotted in figure 4. (The results for $\lambda = 1$ [9] are also plotted for comparison.) As can be seen, the fixed points of equation (3.3) agree very well with the numerical simulations for the radius of attraction when $\lambda = 2$. (Note also that for $\lambda = 1$, equation (3.3) becomes a very good predictor of R for large α where $\rho(t)$ is much more restricted.)

Although it would certainly be desirable to have results for larger values of λ, such simulations become increasingly costly for fully connected networks near saturation, i.e., $P \sim O(N^\lambda/\lambda!)$, in terms of both computer time, growing like $O(N^{2\lambda+1}/(\lambda!)^2)$, and

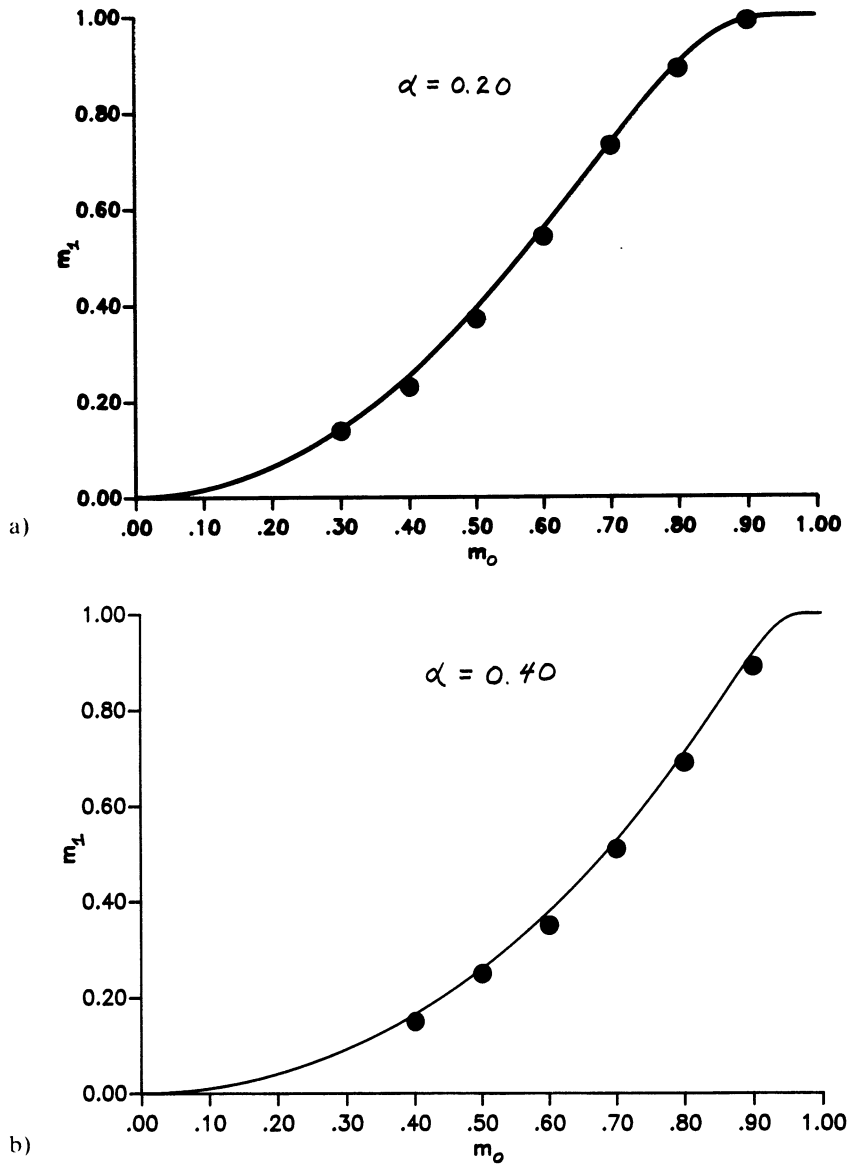

Fig. 3. — a) m_0 *vs.* m_1 at $\alpha = 0.20$. The solid line is a plot of equation (3.3). b) m_0 *vs.* m_1 at $\alpha = 0.40$. The solid line is a plot of equation (3.3). All the data was taken on a network with $N = 64$ neurons.

memory, growing like $O(N^{\lambda}/\lambda!)$. Hence, these simulations must await the next generation of parallel computers.

Discussion.

It has been shown that networks with many-neuron interactions do not store information more efficiently than networks with only two-neuron interactions, and in fact the storage capacity is given exactly by the same Gardner formula. This leads one to speculate on the

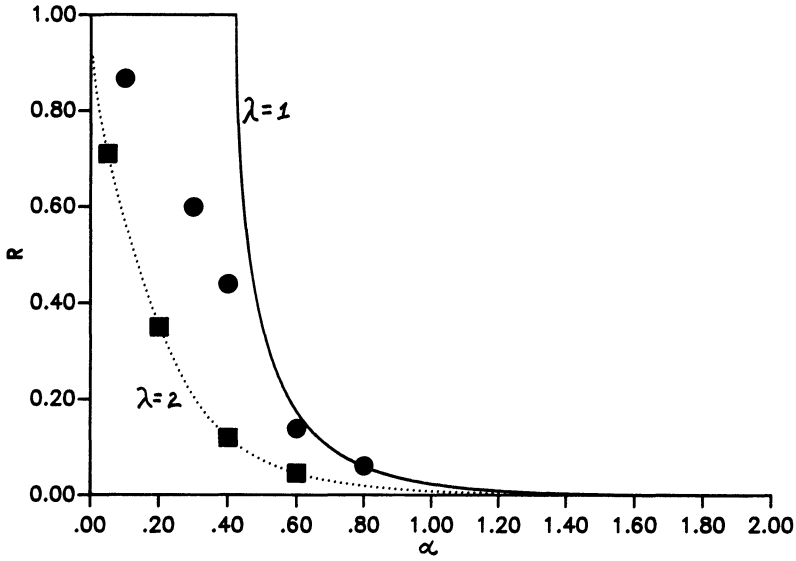

Fig. 4. — The radius of attraction for $\lambda = 1$ (○) and $\lambda = 2$ (□). For $\lambda = 1$ the network size was $N = 100$ while for $\lambda = 2$, the network size was $N = 64$. Also shown are the theoretical predictions.

global nature of the limiting value of $\alpha = 2$. Although the storage capacity of the original *Hopfield model* can be improved upon by deleting connections [16], or by using Ising (± 1) variables for the synapses [17], or by using restricted range interactions [18], none of these approaches leads to increasing α above 2 as long as random uncorrelated patterns are used. It would be interesting to know if α is limited simply by the two state nature of the spin variables or by some more fundamental mechanism. Of course, one can consider correlated patterns in which case the storage capacity defined above goes to infinity as all of the patterns become perfectly correlated ; however, if one then looks at the quantity : *information/synaptic coupling,* this decreases as the correlation between the patterns increases [5], implying once again some fundamental limit to the information storage capacity of neural networks.

On the other hand, many-neuron interacting networks do have a different dynamical behaviour as indicated by equation (3.3). This solution of the dynamics indicates that many-neuron interacting neural networks are far better at pattern discrimination than two-neuron interacting networks, i.e., when operating at a given radius of attraction many-neuron interacting networks are able to store more patterns than the simple two-neuron interacting networks. For example, a reasonable radius of attraction for the purpose of associative memory is $R = 0.20$ (10 % wrong spins). Operating at this value of R, networks with two-neuron interactions can store approximately $0.52\,N$ patterns, while a network with three-neuron interactions can store approximately 0.35 ($N^2/2!$) patterns and a network with four-neuron interactions can store 0.18 ($N^3/3!$) patterns. Hence, the real power of networks with many-neuron interactions will become apparent in applications where it is necessary to obtain a predetermined level of discrimination between large numbers of patterns. (For real world applications this will be important only if the cost of making many-neuron interactions is not significantly more than that for making two-neuron interactions. Such is certainly the case in neurobiology and may help explain nature's propensity for many-neuron connections.)

Another feature of the dynamics obvious from equation (3.3) is the extremely poor behaviour of the network as the limit $\alpha = 2$ is approached. R approaches zero exponentially

fast thus making the network useless for most associative memory applications independent of the value of λ. Instead of working at large α then, these results imply that it would be better to go to a larger value of λ and then work at a smaller value of α.

Finally, it should be added that the calculations presented above can be extended to include networks of mixed type, i.e., networks having for example two-body and three-body interactions, etc., and to networks having less than full connectivity. In such cases, the efficiency of information storage is unchanged from that above, but the dynamics undergoes predictable changes depending upon the exact architecture one is considering.

Acknowledgments.

I would like to thank the theory group at the University of Bonn for their hospitality throughout the earlier phases of this work and the HLRZ for its hospitality while the computer calculations were in progress.

References

[1] PERETTO P. and NIEZ J. J., *Biol. Cybern.* **54** (1986) 53.
[2] PERSONNAZ L., GUYON I. and DREYFUS G., *Europhys. Lett.* **8** (1987) 863 ;
GARDNER E., *J. Phys.* **A 20** (1987) 3453 ;
ABBOTT L. F. and YAIR ARIAN *Phys. Rev.* **A 36** (1987) 5091.
[3] BALDI P. and VENKATESH S. S., *Phys. Rev. Lett.* **58** (1987) 913.
[4] VENKATESH S. S., Neural Networks for Computing, Ed. J. S. Denker (American Inst. of Phys., New York) 1986.
[5] GARDNER E., *Europhys. Lett.* **4** (1987) 481 ; *J. Phys.* **A 21** (1988) 257 ;
GARDNER E. and DERRIDA B., *J. Phys.* **A 21** (1988) 271.
[6] KEPLER T. B. and ABBOTT L. F., *J. Phys. France* **49** (1988) 1657 ;
KRAUTH W., MÉZARD M. and NADAL J.-P., *Complex Syst.* **2** (1988) 387.
[7] WEISBUCH G., *J. Phys. Lett. France* **46** (1985) L623 ;
KRAUTH W., NADAL J.-P. and MÉZARD M., *J. Phys.* **A 21** (1988) 2995.
[8] FORREST B. M., *J. Phys.* **A 21** (1988) 245.
[9] KRÄTZSCHMAR J. and KOHRING G. A. (to be published).
[10] KINZEL W. and OPPER M., Physics of Neural Networks, Eds. J. L. van Hemmen, E. Domany and K. Schulten (Springer Verlag ; to be published).
[11] KOHRING G. A., *Europhys. Lett.* **8** (1989) 697.
[12] GARDNER E., DERRIDA B. and MOTTISHAW P., *J. Phys. France* **48** (1987) 741.
[13] MEIR R. and DOMANY E., *Phys. Rev. Lett.* **59** (1987) 359 ; *Phys. Rev.* **37** (1988) 608.
[14] KRAUTH W. and MÉZARD M., *J. Phys.* **A 20** (1987) L745 ;
ABBOTT L. F. and KEPLER T. B., *J. Phys.* **A 22** (1989) L711.
[15] KANTER I. and SOMPOLINSKY H., *Phys. Rev.* **35** (1987) 380.
[16] CANNING A. and GARDNER E., *J. Phys.* **A 21** (1988) 3275 ;
DERRIDA B., GARDNER E. and ZIPPELIUS A., *Europhys. Lett.* **4** (1987) 167.
[17] AMALDI E. and NICOLIS S., *J. Phys. France* **50** (1989) 2333 ;
KRAUTH W. and OPPER M., *J. Phys.* (to be published) ;
KRAUTH W. and MÉZARD M., Laboratoire de Physique Statistique de l'Ecole Normale Supérieure preprint 95, 1989.
[18] FORREST B. M., *J. Phys. France* **50** (1989) 2003.