*J. Phys. France* **49** (1988) 1657-1662 OCTOBRE **1988,** PAGE 1657

Classification
*Physics Abstracts*
75.10H — 87.30G — 89.70

# Domains of attraction in neural networks

Thomas B. Kepler and L. F. Abbott (*)

Physics Department, Brandeis University, Waltham, MA 02254, U.S.A.

(*Reçu le 28 mars 1988, accepté le 13 juin 1988*)

**Résumé.** — Nous calculons le domaine d'attraction du point fixe d'une mémoire à réseau de neurones comme fonction des champs magnétiques locaux. Ce résultat, combiné à des algorithmes d'apprentissage standard, rend possible la construction de mémoires associatives saturées avec des propriétés de souvenir précisément spécifiées.

**Abstract.** — The domain of attraction of a neural network memory fixed point is computed as a function of its local magnetic fields. When combined with standard learning algorithms, the result makes it possible to construct saturated associative networks memories with precisely specified recall properties.

## Introduction.

A neural network consists of a large number of simple degrees of freedom which exhibit complex and interesting dynamics due to the highly interconnected nature of their couplings. Typically, $N$ Ising variables $S_i = \pm 1$ are coupled through an arbitrary matrix $J_{ij}$. The state of the $S_i$ at time step $t+1$ is given in terms of the state at time $t$ by the simple updating rule,

$$S_i(t+1) = \text{sign}\left(\sum_{j=1}^{N} J_{ij}\, S_j(t)\right). \qquad (1.1)$$

Rule (1.1) corresponds to the parallel dynamics we will consider here (although we will address serial dynamics briefly at the end). A key issue in neural network research is whether by cleverly choosing the matrix $J_{ij}$ we can make the map (1.1) do something interesting and useful.

Associative memory is a task ideally suited to network dynamics. In an associative network we demand that inputs $S_i(0)$ be mapped to outputs $\xi_i$ to which they are sufficiently closely associated. That is, if the input $S_i(0)$ has a large enough overlap

$$m_0 = \frac{1}{N}\sum_{i=1}^{N} S_i(0)\,\xi_i \qquad (1.2)$$

with the memory state $\xi_i$ then the dynamics should produce the final state, for large $t$, $S_i(t) = \xi_i$. This is done by making $\xi_i$ a fixed point of the transformation (1.1) with a large enough domain of attraction to insure that all input states with sufficiently large $m_0$ will be drawn to the fixed point by the network dynamics. Of course to make a memory device there must be many different fixed points corresponding to different memory states. In order to construct a useful network associative memory we must be able to find a matrix having fixed points at the desired memory state locations with domains of attraction appropriately adjusted to include all input states we wish to have mapped to a given fixed point. How can we construct such a matrix ?

A major step towards answering this question has been provided by the Edinburgh group [1, 2]. For a given memory fixed point $\xi_i$ let us define

$$\gamma_i = \frac{1}{\sqrt{N}}\sum_{j=1}^{N} J_{ij}\,\xi_i\,\xi_j\,. \qquad (1.3)$$

In order to specify our normalization we will assume throughout that $J_{ij}$ satisfies the normalization constraint

$$\sum_{j=1}^{N} (J_{ij})^2 = N \qquad (1.4)$$

and in addition we take

$$J_{ii} = 0 \qquad (1.5)$$

(*) Research supported by Department of Energy Contract AC02-ER0320.

for all $i$. In order for the configuration $\xi_i$ to be a fixed point of the transformation (1.1) it is necessary and sufficient that all the $\gamma_i$ be positive. The Edinburgh group has provided a simple algorithm which will construct a matrix with a set of fixed points at any desired locations satisfying the condition

$$\gamma_i \geqslant \kappa \tag{1.6}$$

for positive $\kappa$. The algorithm consists of repeatedly adding a term proportional to $\xi_i \xi_j$ to elements of the matrix $J_{ij}$ for any $\gamma_i < \kappa$ until no such terms remain. The algorithm is known to converge if such a matrix exists and furthermore, Gardner [2] has determined under what conditions the matrix does in fact exist. Therefore, it is possible to construct a network with fixed points at desired locations and with a specified distribution of values $\gamma_i$.

Unfortunately, the above outlined program does not completely answer the questions raised in constructing an associative memory. We still must address the issue of adjusting the domains of attraction of the fixed points. How is the domain of attraction of a fixed point related to the distribution of its $\gamma_i$ values ? More precisely, if we know the distribution of $\gamma_i$ for a fixed-point $\xi_i$ can we predict whether or not a given input with initial overlap $m_0$ as defined in (1.2) will be mapped to the fixed point by the network dynamics ? Our purpose here is to answer this question.

**Calculation of domain of attraction.**

A solution for the time evolution of a state under the map (1.1) has been given for a Hebb type matrix [3] but the analysis is too involved and cumbersome to be of practical value here and we need to allow for a general matrix $J_{ij}$. Our ability to arrive at an answer to the question of domains of attraction rests instead on an empirical observation. Numerical simulation of trained networks near saturation has convinced us that the first step of a parallel dynamics is a very sensitive indicator of the final outcome of that dynamics. Consider a state $S_i(0)$ with initial overlap $m_0$ with a given memory fixed point $\xi_i$ and suppose that after one parallel update (1.1) it reaches a new state $S_i(1)$ with overlap

$$m_1 = \frac{1}{N} \sum_{i=1}^{N} S_i(1)\, \xi_i \,. \tag{2.1}$$

The quantity $m_1 - m_0$ is a measure of the distance the state has travelled in one update of the network. The distance from the initial state $S_i(0)$ to the memory state $\xi_i$ is $1 - m_0$. What we have observed is that if the state travels half the distance to the fixed point or more on its first update then it will ultimately reach the fixed point. On the other hand if it travels less than halfway to the fixed point it will not reach that memory state. In other words, the probability that an input having initial overlap $m_0$ with a given memory fixed point gets mapped to that fixed point is given by

$$P_{S_i \to \xi_i}(m_0) = \theta\left( \frac{m_1 - m_0}{1 - m_0} - \frac{1}{2} \right) . \tag{2.2}$$

The evidence for this surprising result is given in figures 1-4. Figire 1 is a typical histogram showing the fractional number of states mapped to a given memory fixed point as a function of $(m_1 - m_0)/(1 - m_0)$. This figure is based on a 200 node network trained using the Edinburgh algorithm and on 1 000 initial inputs with widely varying initial overlaps. Clearly the theta function is a good approximation of the distribution shown in this figure although the actual curve is somewhat rounded over. Figure 2 shows evidence that this rounding diminishes as a function of $N$. In figure 2 we plot the thickness of

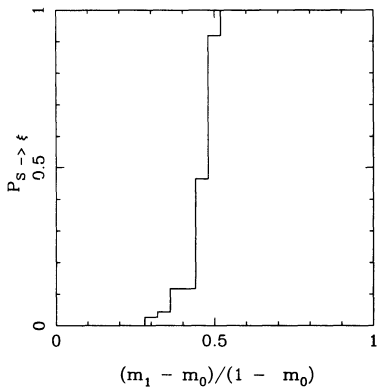


Fig. 1. — A histogram showing the probability that an initial state with overlap $m_0$ is mapped to the fixed point as a function of $(m_1 - m_0)/(1 - m_0)$. The graph is based on a 200 node network and on 1 000 initial points.

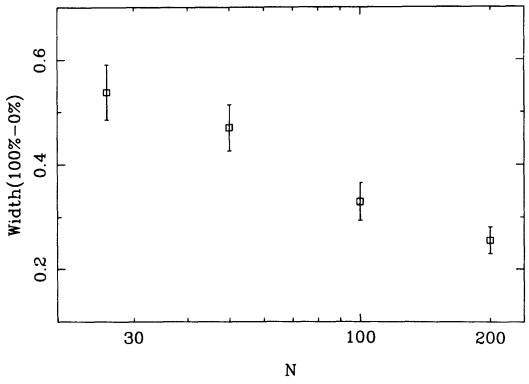


Fig. 2. — The width of the sharp rise in the probability distribution between 0 and 1 plotted as a function of $N$, the size of the network. The shrinking width suggests an approach to a true theta function for large $N$.

the rising part of the distribution as a function of $N$. The points seem to indicate an approach to a real theta function for sufficiently large $N$. The simplicity and universality of this result is remarkable. Figure 3 shows that the point at which the probability reaches one is independent of the value of $\kappa$ associated with the fixed point and figure 4 shows that it is relatively independent of $N$. The large $N$ extrapolation of figure 4 supports the value of 1/2 we have assumed although we do not have a particularly accurate value for the exact location of the step in the theta function and furthermore this may depend on details of the learning algorithm.

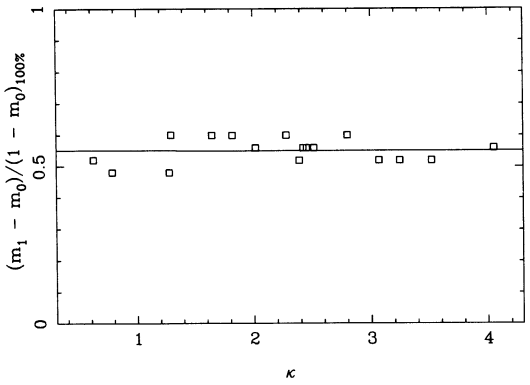


Fig. 3. — The value of $(m_1 - m_0)/(1 - m_0)$ at which all initial inputs get mapped to the memory fixed point plotted against the value of $\kappa$ associated with the fixed point. This shows that the probability distribution $P_{S \to \xi}$ expressed as a function of $(m_1 - m_0)/(1 - m_0)$ is insensitive to the size of the domain of attraction of the memory state. These data correspond to a 100 node network.

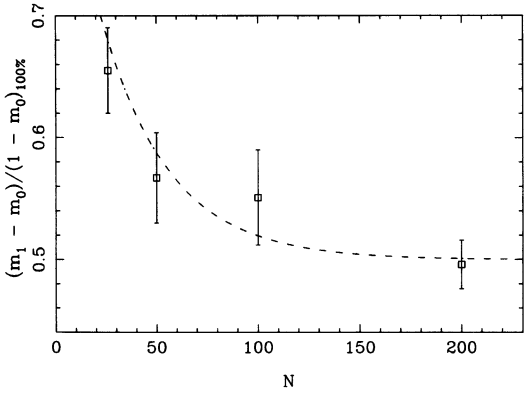


Fig. 4. — The value of $(m_1 - m_0)/(1 - m_0)$ at which all initial inputs are mapped to the memory fixed point for various values of $N$.

The size of the domain of attraction for a memory fixed point clearly depends on the value of $\kappa$ used in the training algorithm, a relation first analysed by Forrest [4]. Equation (2.2) indicates that, quite surprisingly, for the saturated networks we have investigated this $\kappa$ dependence enters solely through the value of $m_1$. In a recent preprint Krauth, Nadal and Mézard [5] have shown that in general the size of the domain of attraction also depends on the symmetry properties of the matrix $J_{ij}$. It should be stressed that the result (2.2) has been obtained by studying networks near saturation, that is, networks with nearly the maximum number of memories for a given $\kappa$ value and that simulations were done for $\kappa$ always greater than 3/4 and most often greater than one. In our training procedure we started with a symmetric matrix and then applied the Edinburgh algorithm. As a result the final trained matrices were very close to being symmetric. Thus, our simulations have not probed networks far from saturation or studied the effects of asymmetry. The result (2.2) exhibits no dependence on the symmetry properties of the matrix $J_{ij}$ because $m_1$ is independent of asymmetry. Krauth, Nadal and Mézard have studied networks very far from saturation, in particular, networks with a single memory state. Since for these networks asymmetry does have an important impact on domains of attraction it is clear that (2.2) cannot be correct for all values of $\kappa$ and over the entire range of allowed numbers of memory states. It is particularly suspect for networks far from saturation. However, it is valid for networks near saturation and $\kappa$ greater than 3/4 where we will use it here. We are currently studying the possibility of extending this rule to include asymmetry dependence.

Once we realize how good an indicator the first step dynamics is for the final outcome of a network map, we can compute the first step dynamics and provide a prediction for the domain of attraction of a fixed point characterized by a set of $\gamma_i$'s. The probability distribution for first step overlaps $m_1$ given an initial overlap $m_0$ is

$$P(m_1|m_0) = \frac{\mathrm{Tr}_S\left[\delta\left(m_1 - \frac{1}{N}\sum_{i=1}^{N}\xi_i \,\mathrm{sign}\left(\sum_{j=1}^{N} J_{ij} S_j\right)\right)\delta\left(m_0 - \frac{1}{N}\sum_{j=1}^{N} S_j \xi_j\right)\right]}{\mathrm{Tr}_S\left[\delta\left(m_0 - \frac{1}{N}\sum_{j=1}^{N} S_j \xi_j\right)\right]} . \tag{2.3}$$

It is convenient to introduce the fields

$$h_i = \frac{1}{\sqrt{N}}\sum_{j=1}^{N} J_{ij}\,\xi_i\,S_j \tag{2.4}$$

and to write

$$P(m_1|m_0) = \int \prod_i \mathrm{d}h_i \, \mathcal{P}(\{h_i\}|m_0) \, \delta\left(m_1 - \frac{1}{N}\sum_{i=1}^{N} \mathrm{sign}\,(h_i)\right) . \tag{2.5}$$

We will assume that the rows of the matrix $J_{ij}$ are uncorrelated so that the $h_i$ act as independent variables. This is true of matrices trained with the Edinburgh algorithm if the memory states are independent. Thus,

$$\mathcal{P}(\{h_i\}|m_0) = \prod_{i=1}^{N} p(h_i|m_0) \tag{2.6}$$

with

$$p(h_i|m_0) = \frac{\mathrm{Tr}_S\left[\delta\left(m_0 - \frac{1}{N}\sum_{j=1}^{N} S_j\,\xi_j\right)\,\delta\left(h_i - \frac{1}{\sqrt{N}}\sum_{j=1}^{N} J_{ij}\,\xi_i\,S_j\right)\right]}{\mathrm{Tr}_S\left[\delta\left(m_0 - \frac{1}{N}\sum_{j=1}^{N} S_j\,\xi_j\right)\right]} . \tag{2.7}$$

Introducing an integral representation for the delta functions and taking the trace over configurations $S_i$, the numerator of equation (2.7) is proportional to

$$\int \mathrm{d}x\,\mathrm{d}y \exp\left[iyh_i + iNxm_0 + \sum_j \ln\cos\left(\frac{1}{\sqrt{N}}\,\xi_j\,x + J_{ij}\,\xi_i\,y\right)\right] . \tag{2.8}$$

The cosine term can be expanded in powers of $1/N$ to give

$$\sum_j \left[\ln\cos\,(\xi_j\,x) - \frac{1}{\sqrt{N}}J_{ij}\,\xi_i\,\xi_j\,y\tan\,(x) - \frac{1}{2N}J_{ij}^2\,y^2(1+\tan^2\,(x))\right] . \tag{2.9}$$

This expression can be further simplified by using the normalization condition (1.4) and the definition of $\gamma$, (1.3). The $x$ integration in (2.7) can be done by saddle-point approximation at the stationary point

$$\tan x = im_0 . \tag{2.10}$$

Finally the $y$ integration is an ordinary Gaussian integral giving

$$p(h_i|m_0) = \frac{1}{\sqrt{2\,\pi\,(1-m_0^2)}}\exp -\left(\frac{(h_i-\gamma_i)^2}{2(1-m_0^2)}\right) . \tag{2.11}$$

We can now evaluate the desired quantity $P(m_1|m_0)$ from equation (2.5). Using an exponential representation of the delta function and performing the $h_i$ integrations we find

$$P(m_1|m_0) = \int \mathrm{d}x \exp\left(im_1\,x - \sum_i \ln\left[\cos\left(\frac{x}{N}\right) - i\sin\left(\frac{x}{N}\right)\mathrm{erf}\left(\frac{m_0\,\gamma_i}{\sqrt{2(1-m_0^2)}}\right)\right]\right) . \tag{2.12}$$

Expanding for large $N$ we obtain the simple result

$$P(m_1|m_0) = \delta\left(m_1 - \frac{1}{N}\sum_i \mathrm{erf}\,(m_0\,\gamma_i/\sqrt{2(1-m_0^2)})\right) . \tag{2.13}$$

In other words, for every matrix satisfying (1.4) and (1.6) we find that after one parallel iteration an initial state with overlap $m_0$ has a definite overlap $m_1$ determined solely by the distribution of $\gamma_i$'s at the fixed point. If we know this normalized distribution, $\rho(\gamma)$ then (this result was also obtained in Ref. [4])

$$m_1 = \int \mathrm{d}\gamma\,\rho(\gamma)\,\mathrm{erf}\,(m_0\,\gamma/\sqrt{2(1-m_0^2)}) . \tag{2.14}$$

Note that this result is self-averaging. It is true for all matrices satisfying (1.4) and (1.6) and was obtained without averaging over matrices $J_{ij}$.

Finally using our observations about the form of the probability function (2.2) we can use the result just obtained to give a prediction for the domain of attraction of a fixed point memory state characterized by a distribution $\rho(\gamma)$. The fixed point will attract all states with initial overlap larger than a

value $m_c$ determined by the condition that $(m_1(m_c) - m_c)/(1 - m_c) = 1/2$ Thus, the domain of attraction is given by the equation

$$m_c + 1 = 2 \int d\gamma\, \rho(\gamma)\, \mathrm{erf}\,(m_c\, \gamma / \sqrt{2(1 - m_c^2)})\,. \tag{2.15}$$

**Calculation of the distribution function.**

Before we can use equation (2.15) we must know the distribution function $\rho(\gamma)$. This can of course be determined in practice by using the definition (1.3) but it is useful to see how it is related to the value of $\kappa$ which characterizes the fixed point (Eq. (1.6)). We are ultimately interested in $\rho(\gamma)$ for an arbitrary matrix satisfying (1.4) and (1.6) for a set of $\alpha N$ memory fixed points. However, in order to compare our results with previous work we will first calculate $\rho(\gamma)$ for a Hebb type matrix.

Suppose $J_{ij}$ takes the Hebb form

$$J_{ij} = \frac{1}{\sqrt{\alpha N}} \sum_{\mu=1}^{\alpha N} \xi_i^\mu\, \xi_j^\mu \tag{3.1}$$

and we wish to compute the $\gamma$ distribution for one of the memory states, say $\xi_i^1$. We do this by averaging over all possible states $\xi_i^\mu$ and we find

$$\rho(\gamma) = \exp -\frac{1}{2}\left(\gamma - \frac{1}{\sqrt{\alpha}}\right)^2. \tag{3.2}$$

When combine with equation (2.14) this gives the prediction for first step dynamics

$$m_1 = \int \frac{d\gamma}{\sqrt{2\pi}} \exp -\frac{1}{2}\left(\gamma - \frac{1}{\sqrt{\alpha}}\right)^2 \times \times \mathrm{erf}\,(m_0\, \gamma / \sqrt{2(1 - m_0^2)}) \tag{3.3}$$

Gardner, Derrida and Mottishaw [3] have given the expression

$$m_1 = \mathrm{erf}\,(m_0 / \sqrt{2\,\alpha}) \tag{3.4}$$

in their analysis of network dynamics for a Hebb type matrix. Although the two expressions look quite different they can be shown to be equivalent first by noting that they agree for $m_0 = 0$ and then by establishing the equality of their derivatives with respect to $m_0$ for all $m_0$.

The calculation of $\rho(\gamma)$ for a general matrix $J_{ij}$ can also be done (see also Ref. [6]). Assuming that the matrix $J_{ij}$ satisfies the stability condition (1.6) for a set of $\alpha N$ independent memory states, we choose arbitrarily to examine the $\gamma$ distribution associated with the state $\xi_i^1$ at the site $i$ since all states and all sites are equivalent.

We wish to average over all memory states $\xi_i^\mu$ and over all matrices $J_{ij}$ subject to the constraints (1.4) and (1.6). The distribution function is defined by

$$\rho(\gamma) = \left\langle \frac{1}{\mathcal{N}} \int \prod_j dJ_{ij} \prod_{\mu=1}^{N} \theta\left(\frac{1}{\sqrt{N}} \sum_j J_{ij}\, \xi_i^\mu\, \xi_j^\mu - \kappa\right) \delta\left(\sum_j (J_{ij}^a)^2 - N\right) \delta\left(\frac{1}{\sqrt{N}} \sum_j J_{ij}\, \xi_i^1\, \xi_j^1 - \gamma\right)\right\rangle \tag{3.5}$$

where

$$\mathcal{N} = \int \prod_j dJ_{ij}\, \delta\left(\sum_j (J_{ij})^2 - N\right) \prod_{n=1}^{N} \theta\left(\frac{1}{\sqrt{N}} \sum_j J_{ij}\, \xi_i^\mu\, \xi_j^\mu - \kappa\right). \tag{3.6}$$

Here the angle brackets denote an average over memory configurations $\xi_i^\mu$. In order to perform the average over the quantity $1/\mathcal{N}$ we introduce replicas $J_{ij}^a$ with $a = 1, 2, \ldots, n$ and write

$$1/\mathcal{N} = \lim_{n \to 0} \mathcal{N}^{n-1}. \tag{3.7}$$

The calculation from this point on is very similar to the one performed in reference [2] so we will not supply the details here. A mean field variable representing the overlap between replicas

$$q_{ab} = \frac{1}{N} \sum_j J_{ij}^a\, J_{ij}^b \tag{3.8}$$

is introduced and plays a crucial role. We assume that the replica symmetric solution

$$q_{ab} = q \tag{3.9}$$

is appropriate. Then the result for the averaged generating function is

$$\rho(\gamma) = \int \frac{dz}{\pi \sqrt{q(1-q)}} e^{-\frac{z^2}{2q}} \times \times \left[\frac{e^{-\frac{(\gamma - z)^2}{2(1-q)}}\, \theta(\gamma - \kappa)}{1 - \mathrm{erf}\left(\frac{\kappa - z}{\sqrt{2(1-q)}}\right)}\right]. \tag{3.10}$$

When we saturate the memory by going to the maximum value of $\alpha$ allowed for a given $\kappa$ then

$q \to 1$ [2]. In this limit the distribution function simplifies to

$$\rho(\gamma) = \frac{e^{-\frac{\gamma^2}{2}}}{\sqrt{2\pi}}\,\theta(\gamma - \kappa) + \\ + \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{\kappa}{\sqrt{2}}\right)\right]\delta(\gamma - \kappa)\,. \quad (3.11)$$

In figures 5 and 6 we have used such a saturated distribution to show our results. Figure 5 compares our prediction (2.14) with computer generated data for several values of $\kappa$. The final result of our work is show in figure 6 where we plot $m_c$ as a function of $\kappa$ for saturated networks. The computer data agree well with our predictions.

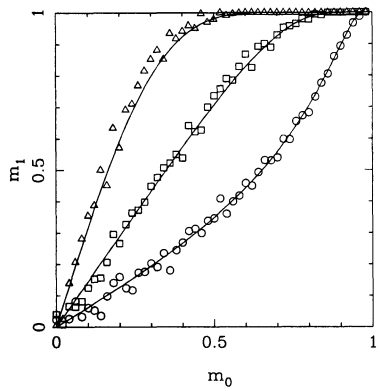


Fig. 5. — The predicted value of $m_1$ as a function of $m_0$ compared with computer data for three different distributions $\rho(\gamma)$. Going from the top curve to the bottom these correspond to $\kappa$ values of 4.05, 2.02 and 0.78.

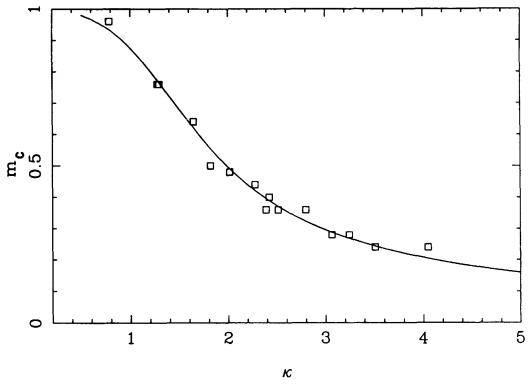


Fig. 6. — The predicted value for the domain of attraction $m_c$ as a function of $\kappa$ for saturated networks compared with computer generated data.

## Conclusions.

The results shown in figure 6 determine the value of $\kappa$ needed in a saturated network to produce a desired domain of attraction $m_c$. The work of Gardner [2] determines whether or not a matrix with the desired properties exists and finally the Edinburgh algorithm [1] and a sufficient amount of computer time allow for the construction of a matrix with precisely the properties desired. We can and have also treated correlated memory states and initial inputs that overlap with more than one of the memory fixed points. This involves a straightforward extension of the techniques applied here. Our results seem to apply at least approximately to serial dynamics as well. In addition, our prediction for the size of the domain of attraction applies individually to each independent fixed point so by appropriately adjusting the $\gamma$ distribution for each memory state we can individually adjust each domain.

## Acknowledgments.

We wish to thank D. Amit, H. Gutfreund and M. Mézard for helpful discussions and are grateful to M. Mézard and E. Gardner for pointing out an error in an earlier version of this paper.

## References

[1] WALLACE, D. J., *Advances in Gauge Theory* Eds D. W. Duke and J. F. Owens (World Scientific, Philadelphia) 1985 ;
BRUCE, A. D., GARDNER, E. and WALLACE, D. J., *J. Phys. A* **20** (1987) 2909.

[2] GARDNER, E., *J. Phys. A* **21** (1988) 257.

[3] GARDNER, E., DERRIDA, B. and MOTTISHAW, P., *J. Phys. France* **48** (1987) 741.

[4] FORREST, B., *J. Phys. A* **21** (1988) 245.

[5] KRAUTH, W., NADAL, J.-P. and MÉZARD, M., Ecole Normale Supérieure, preprint (1988).

[6] KRAUTH, W., MÉZARD, M. and NADAL, J.-P., Ecole Normale Supérieure, preprint (1988).