Neural Networks, Vol. 3, pp. 613–623, 1990
Printed in the USA. All rights reserved.

0893-6080/90 $3.00 + .00

ORIGINAL CONTRIBUTION

Shaping Attraction Basins in Neural Networks

SANTOSH S. VENKATESH AND GIRISH PANCHA

Moore School of Electrical Engineering, University of Pennsylvania

DEMETRI PSALTIS

Department of Electrical Engineering, California Institute of Technology

AND GABRIEL SIRAT

Groupe Optice de Materiele, École Nationale Superieure de Telecommunication

(*Received* 19 *January* 1989; *revised and accepted* 22 *February* 1990)

Abstract—*An interesting duality between two formally related schemes for neural associative memory is exploited to shape the attraction basins of stored memories. Considered are a family of spectral algorithms—based on specifying the spectrum of the matrix of weights as a function of the memories to be stored—and a class of dual spectral algorithms—based on manipulations of the orthogonal subspace of the memories, which are expanded here. These algorithms are shown to attain near maximal memory storage capacity of the order of n, and are shown to typically require the order of n^3 elementary operations for their implementation. Signal-to-noise ratio arguments are presented showing a duality in the error-correction behaviour of the two schemes: the spectral algorithm demonstrates memory-specific attraction around the memories, while the dual spectral algorithm demonstrates direction-specific attraction. Composite algorithms capable of joint memory-specific and direction-specific attraction are presented as a means of variably shaping attraction basins around desired memories. Computer simulations are included in support of the analysis.*

Keywords—Associative memory, Network dynamics.

1. INTRODUCTION

In this paper we develop the duality between two methods for training a fully connected network of n McCulloch–Pitts neurons (McCulloch & Pitts, 1943). The sum of outer products is perhaps the most often used training method for such networks (Nakano, 1972; Amari, 1977; Hopfield, 1982). The memory storage capacity for this method is $n/4 \log n$ (McEliece, Posner, Rodermich, & Venkatesh, 1987; Psaltis & Venkatesh, 1989) whereas the maximal theoretical capacity for any storage algorithm is $2n$ (Cover, 1965; Venkatesh, 1986b). The spectral algorithm (Kohonen, 1977; Personnaz, Guyon, & Dreyfus, 1985; Venkatesh & Psaltis, 1989) and an algorithm we will refer to as the dual spectral algorithm (Maruani, Chevallier, & Sirat, 1987) are algorithms whose capacities approach the theoretical maximum. In this paper, we briefly review these two algorithms, establish the relationship between them, and define how a proper choice of parameters specifies their error correction properties.

In such networks, memories to be stored are typically programmed as fixed points of the structure. Error correction is obtained by attracting to one of the stored fixed points, initial states (or probes) of the system that are close to the fixed points. We show that in the spectral scheme the radius of attraction around each of the stored stable states is controlled by the relative size of the eigenvalues of the interconnection matrix. The dual spectral algorithm, on the other hand, leads to a method for programming the shape of the attraction basin around each of the elements of the stored vectors. We present a new method based on linear programming for selecting the parameters of the dual spectral algorithm which determine its attraction dynamics around each stored fixed point and we suggest a hybrid algorithm that can provide more arbitrary control of the shape of the attraction basin.

We consider a fully interconnected network of n McCulloch–Pitts neurons with the instantaneous bi-

Acknowledgement: The work of the first two authors was supported in part by NSF grant EET-8709198. The work at Caltech is supported by DARPA and AFOSR.

Requests for reprints should be sent to Demetri Psaltis, Department of Electrical Engineering, Caltech, MS-116-81, Pasadena, CA 91125.

nary outputs (-1 or 1) of each of the neurons being fed back as inputs to the network: if $u_1[t]$, $u_2[t]$, . . . , $u_n[t]$ are the outputs of each of the n neurons in the network at epoch t, then the neural update of the ith neuron results in a new state at epoch $t + 1$ according to the familiar threshold rule:

$$u_i[t + 1] = \Delta\left(\sum_{j=1}^{n} w_{ij}u_j[t] - w_{i0}\right),$$

where

$$\Delta(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0. \end{cases}$$

The mode of operation may be synchronous (with all the neurons being updated simultaneously at each epoch) or asynchronous (with at most one neuron being updated at each epoch). In the application of these networks to associative memory both modes of operation lead to very similar associative behaviour (cf. Psaltis & Venkatesh, 1989, for instance) and we will not make a distinction in this paper as to the precise mode of operation.

The nature of flow in state space is completely determined once the neural interconnection strengths and the mode of operation is specified. We will be interested in specifying patterns of interconnectivity for which arbitrarily prescribed m-sets of memories $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)} \in \mathrm{B}^n$ can be stored in the network. In order for the network to act as an associative memory, we require that the memories themselves be stable (i.e., all subsequent operations on the memory $\mathbf{u}^{(\alpha)}$ give back $\mathbf{u}^{(\alpha)}$). Stable memories are hence fixed points of the network. Furthermore, we require states close to any of the memories to be mapped into the memory by the network. This is the associative or error correcting feature requisite in an associative memory. We call the average Hamming distance from a memory over which such error correction is exhibited the *attraction radius* of the memory.

The quadratic Hamiltonian (energy) and the Manhattan form have been shown to be Lyapunov functions for fully connected networks with symmetric connections (Hopfield, 1982; Goles & Vichniac, 1986; Peretto & Niez, 1986; Psaltis & Venkatesh, 1989), hence, guaranteeing that state trajectories of such networks will terminate in stable points. If the neural interconnection weights are chosen so that the desired memories are stable, then the existence of a Lyapunov function for the system indicates that the memories will exhibit an attraction radius of error correction. The outer product and the dual spectral algorithms lead to symmetric weights but this is not generally true for the spectral scheme. Nevertheless, the spectral scheme also exhibits very similar attraction dynamics (Psaltis & Venkatesh, 1989), even though there is no known Lyapunov function for the general case. In all these algorithms stability of the stored memories can be assured with high probability if the number of memories is within the storage capacity of the algorithm (McEliece et al., 1987; Psaltis & Venkatesh, 1989). The existence of Lyapunov functions then guarantees that the memories (being fixed points) lie at the minima of the Lyapunov functions.

2. ALGORITHMS

2.1. The Spectral Algorithm

In the spectral scheme, the interconnection matrix $\mathbf{W}^s$ is defined as follows:

$$\mathbf{W}^s = \mathbf{U}\Lambda(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T, \tag{1}$$

where $\Lambda = \mathbf{dg}[\lambda^{(1)}, \ldots, \lambda^{(m)}]$ is the $m \times m$ diagonal matrix of positive eigenvalues $\lambda^{(1)}, \ldots, \lambda^{(m)} > 0$, and $\mathbf{U} = [\mathbf{u}^{(1)}\mathbf{u}^{(2)} \cdots \mathbf{u}^{(m)}]$ is the $n \times m$ matrix of memory column vectors.

We note that

$$\mathbf{W}^s\mathbf{U} = \mathbf{U}\Lambda, \tag{2}$$

where $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)}$ are the eigenvectors of $\mathbf{W}^s$ and Λ is the spectrum of $\mathbf{W}^s$ (Venkatesh & Psaltis, 1985; Personnaz, Guyon, & Dreyfus, 1985; Venkatesh & Psaltis, 1989). Therefore, we are guaranteed to have stable memories as long as $\mathbf{W}^s$ is well defined.

For the case of an m-fold degenerate spectrum $\lambda^{(1)}, \ldots, \lambda^{(m)} = \lambda > 0$, we see that the matrix $\mathbf{W}^s$ is symmetric with nonnegative eigenvalues (i.e., it is nonnegative definite). Therefore there exist Lyapunov functions in this case, and moreover it has been shown that the stored memories form *global* energy minima (Venkatesh & Psaltis, 1989).

For the general spectral matrix in eqn (1), exact Lyapunov functions are hard to come by. The signal-to-noise ratio, however, serves as a good ad hoc measure of attraction capability. Consider synchronous operations with $\mathbf{W}^s$ on a state vector $\mathbf{u} = \mathbf{u}^{(\alpha)} + \delta\mathbf{u} \in \mathrm{B}^n$. We have

$$\mathbf{W}^s\mathbf{u} = \mathbf{W}^s(\mathbf{u}^{(\alpha)} + \delta\mathbf{u}) = \mathbf{W}^s\mathbf{u}^{(\alpha)} + \mathbf{W}^s\delta\mathbf{u}.$$

Once again, there exists a "signal" term, $\mathbf{W}^s\mathbf{u}^{(\alpha)}$, and a "noise" term, $\mathbf{W}^s\delta\mathbf{u}$. We anticipate that the greater the signal-to-noise ratio, the greater the attraction around $\mathbf{u}^{(\alpha)}$. Let the Hamming distance between $\mathbf{u}$ and $\mathbf{u}^{(\alpha)}$, $d_H(\mathbf{u}, \mathbf{u}^{(\alpha)})$, equal d (i.e., $\|\delta\mathbf{u}\| = 2\sqrt{d}$). The (strong) norm of the matrix $\mathbf{W}^s$ is defined as

$$\|\mathbf{W}^s\| = \sup_{\mathbf{x}} \frac{\|\mathbf{W}^s\mathbf{x}\|}{\|\mathbf{x}\|}, \quad \|\mathbf{x}\| \neq 0.$$

It follows (cf. Strang, 1980) that $\|\mathbf{W}^s\| = \sqrt{k}$, where k is the largest eigenvalue of the matrix $(\mathbf{W}^s)^T\mathbf{W}^s$. For the case of the degenerate spectrum $\lambda^{(1)}, \ldots, \lambda^{(m)} = \lambda > 0$, $\mathbf{W}^s$ is symmetric, and $(\mathbf{W}^s)^T\mathbf{W}^s = (\mathbf{W}^s)^2$. Therefore, the maximum eigenvalue of

$(\mathbf{W}^s)^T\mathbf{W}^s = k = \lambda^2$, and the signal-to-noise ratio (SNR) is given by

$$\text{SNR} = \frac{\|\mathbf{W}^s\mathbf{u}^{(\alpha)}\|}{\|\mathbf{W}^s\delta\mathbf{u}\|} \geq \frac{\lambda^{(\alpha)}\sqrt{n}}{(\sqrt{k})(2\sqrt{d})} = \frac{1}{2}\sqrt{\frac{n}{d}}.$$

Thus, we would expect the attraction sphere around $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)}$ to increase as n increases for the m-fold degenerate spectral scheme. For the general nondegenerate case, we expect that by varying the size of $\lambda^{(\alpha)}$, the SNR, and hence the attraction capability, be proportionately increased or decreased for the αth memory $\mathbf{u}^{(\alpha)}$ (Figure 1).

Using a result of Komlós (1967) we can show that for all randomly chosen n-tuples $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)} \in \mathbb{B}^n$, and $m \leq n$, the probability that $\mathbf{W}^s$ is well defined approaches one as $n \to \infty$. It immediately follows that the static capacity of the spectral scheme is n, as a linear transformation has at most n eigenvalues.

Let N^s denote the number of elementary operations required to compute the weight matrix $\mathbf{W}^s$ directly from the m memories to be stored. Then using the fact that $(\mathbf{U}^T\mathbf{U})^{-1}$ is symmetric, we can use the Cholesky decomposition to compute its inverse. This along with the rest of the matrix multiplications gives us that $N^s = mn^2 + m^2n + (m^3)/2 + O(n^2)$ (details can be found in Venkatesh and Psaltis (1989)).

2.2. Dual Spectral Algorithms

2.2.1. *Orthogonal Spaces and Duality*. The following scheme, formally related to the outer product and spectral algorithms, was introduced by Maruani et al. (1987).

Let $\mathbf{U} = [\mathbf{u}^{(1)}\mathbf{u}^{(2)} \cdots \mathbf{u}^{(m)}]$ be the matrix of memories as before. Let $\mathbf{x}^{(\beta)}$, $\beta = 1, \ldots, n - m$, be a set of linearly independent vectors in $\mathbb{R}^n$ which are individually orthogonal to each of the memories (i.e., $\mathbf{X}^T\mathbf{U} = \mathbf{0}$, where we define the $n \times (n - m)$ matrix $\mathbf{X} = [\mathbf{x}^{(1)}\mathbf{x}^{(2)} \cdots \mathbf{x}^{(n-m)}]$). Define a weight matrix $\mathbf{W}$ with weights w_{ij} given by

$$w_{ij} = \begin{cases} -\sum_{\beta=1}^{n-m} x_{i\beta}x_{j\beta} & \text{if } i \neq j \\ 0 & \text{if } i = j, \end{cases}$$

where $x_{k\beta}$ is the kth component of $\mathbf{x}^{(\beta)}$. If we define $\hat{\mu}_i = \sum_{\beta=1}^{n-m} x_{i\beta}^2$, $i = 1, \ldots, n$, we see that

$$\mathbf{W} = \hat{\mathbf{M}} - \mathbf{X}\mathbf{X}^T, \tag{3}$$

where $\hat{\mathbf{M}} = \mathbf{dg}[\hat{\mu}_1, \ldots, \hat{\mu}_n]$. Thus,

$$\begin{aligned} \mathbf{W}\mathbf{U} &= \hat{\mathbf{M}}\mathbf{U} - \mathbf{X}\mathbf{X}^T\mathbf{U} \\ &= \hat{\mathbf{M}}\mathbf{U}. \end{aligned} \tag{4}$$

Comparing eqns (2) and (4) we see that the spectral and dual spectral algorithms exhibit an interesting duality. Since the parameters $\hat{\mu}_i$ are positive for each choice of i, it follows that

$$\Delta(\mathbf{W}\mathbf{u}^{\alpha})_i = \Delta(\hat{\mu}_i u_i^{\alpha}) = u_i^{\alpha},$$
$$\text{for each } i = 1, \ldots, n, \ \alpha = 1, \ldots, m.$$

So the memories $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)}$ are fixed points in the scheme as well.

$\mathbf{W}$ as defined in eqn (3) is a zero-diagonal symmetric matrix. Thus, we know that there exists some form of attraction behaviour. However, since the orthogonal basis $\mathbf{X}$ has been chosen arbitrarily, there is some lack of control in specifying attraction capability. Specifically, as we shall argue below, the $\hat{\mu}_i$'s essentially control directional attraction and we have no means of specifying these under the above

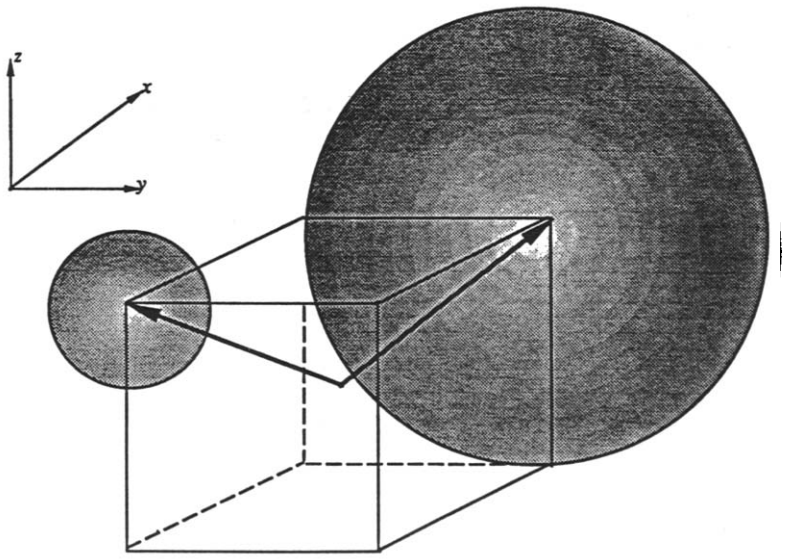

FIGURE 1. Schematic representation of the attraction space in the spectral scheme for memories with different eigenvalues.

approach. Our goal here will be to specify an algorithm where such control is possible.

2.2.2. *The Effect of the μ-values.* In the spectral scheme, the eigenvectors of $\mathbf{W}^s$ are the memories, so that the column space of $\mathbf{W}^s$ is given by the span of the memories. Therefore, if the memories are far enough from each other and the initial state vector $\mathbf{u}$ is close enough to a memory, $\mathbf{W}^s$ combined with the thresholding operation *projects* $\mathbf{u}$ onto the memory.

On the other hand, in the dual spectral scheme, the weight matrix $\mathbf{W}^d$ is obtained by taking the correlation of vectors that are orthogonal to the memories and then setting the diagonal elements to be 0. In creating the zero diagonal, we essentially add perturbations to the left nullspace of $\mathbf{U}$ in the directions of the memories. The *strength* of the perturbations along any component i, is proportional to $\hat{\mu}_i$. Thus, each of the $\hat{\mu}_i$'s corresponds to a directional distortion, and we expect the SNR of the dual spectral scheme to vary from direction to direction proportionately with the value of $\hat{\mu}_i$. We therefore expect that the larger the $\hat{\mu}_i$, more information is lost if the ith bit is flipped and, hence, the smaller the attraction would be in the ith direction.

As an illustration, let us consider the case where $n = 3$, and $\mu_x \ll \mu_y, \mu_z$ (Figure 2). Each memory $\mathbf{u}$ would be preferentially attracted in the dx-direction, indicated schematically by an attraction cone in Figure 2 (i.e., a vector with a different x component will probably map back to $\mathbf{u}$ but vectors with different y and z components will probably not be within the attraction region of $\mathbf{u}$). In other words,

$$\mathbf{P}\left[\begin{pmatrix} -u_x \\ u_y \\ u_z \end{pmatrix} \longrightarrow \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}\right] > \mathbf{P}\left[\begin{pmatrix} u_x \\ u_y \\ -u_z \end{pmatrix} \longrightarrow \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}\right],$$

$$\mathbf{P}\left[\begin{pmatrix} u_x \\ -u_y \\ u_z \end{pmatrix} \longrightarrow \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}\right].$$

2.2.3. *Specifying Directional Attraction With Linear Programming.* The previous section's discussions point to a necessity of somehow specifying the μ-values if we require direction-specific attraction. Specifically, for a *prescribed set* $\mu_1, \ldots, \mu_n > 0$ of directional attraction strengths, and $\mathbf{M} = \mathbf{dg}[\mu_1, \ldots, \mu_n]$, we require a weight matrix $\mathbf{W}^d$ such that

$$\mathbf{W}^d\mathbf{U} = \mathbf{MU}. \tag{5}$$

We define $\mathbf{W}^d$ such that:

$$w_{ij}^d = \begin{cases} -\sum_{\beta=1}^{m} (x_{i\beta}b_\beta)(x_{j\beta}b_\beta) & \text{if } i \neq j \\ 0 & \text{if } i = j, \end{cases} \tag{6}$$

where $x_{i\beta}$ is the ith component of the basis vector $\mathbf{x}^{(\beta)}$ as defined earlier, and b_β is the βth component of a vector which we will specify shortly. Thus, given $\mu_1, \ldots, \mu_n$ we need to find a vector $\mathbf{b}$ such that with $\mathbf{Y} = \mathbf{Xb}$

$$\mathbf{W}^d = \mathbf{M} - \mathbf{YY}^t \tag{7}$$

(Note that the columns of $\mathbf{Y}$, in general, are not orthogonal.)

Assuming that $\mathbf{W}^d$ has the form given in eqn (6), let us now consider the effect of $\mathbf{W}^d$ on the ith ele-

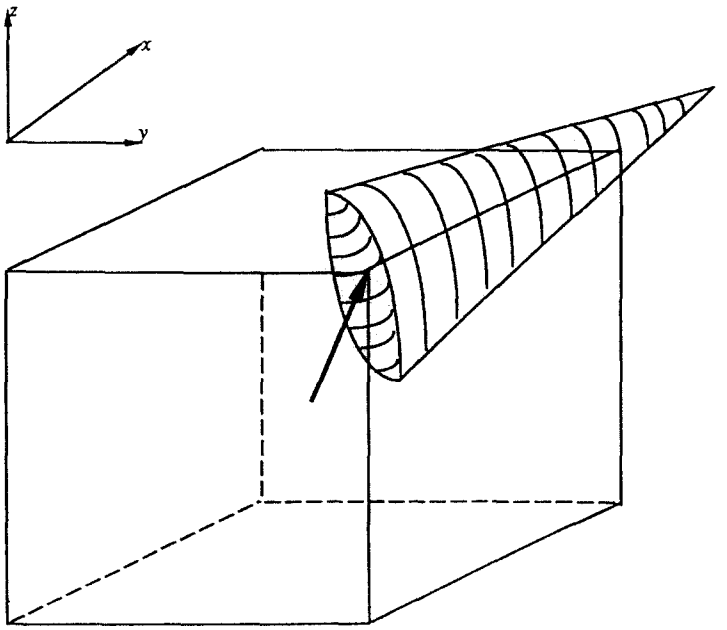

FIGURE 2. Schematic representation of the directional attraction space in the dual spectral scheme for a choice of $\mu_x \ll \mu_y, \mu_z$.

ment of a memory $\mathbf{u}^{(\alpha)}$:

$$[\mathbf{W}^d\mathbf{u}^{(\alpha)}]_i = \sum_{j=1}^{n} w_{ij}^d u_j^{(\alpha)}$$

$$= -\sum_{j\neq i}\sum_{\beta=1}^{n-m} b_\beta^2 x_{i\beta}x_{j\beta}u_j^{(\alpha)}$$

$$= \sum_{j=1}^{n}\sum_{\beta=1}^{n-m} b_\beta^2 x_{i\beta}x_{j\beta}u_j^{(\alpha)} + \sum_{\beta=1}^{n-m} b_\beta^2 x_{i\beta}^2 u_i^{(\alpha)}$$

$$= \sum_{\beta=1}^{n-m} b_\beta^2 x_{i\beta}^2 u_i^{(\alpha)}.$$

We require from eqn (5) that

$$[\mathbf{W}^d\mathbf{u}^{(\alpha)}]_i = \mu_i u_i^{(\alpha)}.$$

where $\mu_i > 0$. By inspection, we obtain the relationship

$$\mu_i = \sum_{\beta=1}^{n-m} x_{i\beta}^2 b_\beta^2.$$

Define $a_{i\beta} = x_{i\beta}^2$, and $c_\beta = b_\beta^2$. Then we require

$$\mathbf{Ac} = \mathbf{M}_\mu,$$

where $\mathbf{A}$ is a known $n \times (n - m)$ matrix with nonnegative elements $a_{i\beta} = x_{i\beta}^2$, $\mathbf{c}$ is an unknown $(n - m)$-dimensional vector with $c_\beta = b_\beta^2$ constrained to be nonnegative, and $\mathbf{M}_\mu$ is a specified n-dimensional vector with positive components $\mu_1, \ldots, \mu_n$.

We notice that this is an overspecified system of n equations with $(n - m)$ unknowns, where both $\mathbf{c}$ and $\mathbf{M}_\mu$ are constrained to have nonnegative elements. Linear programming techniques can be used to solve this system of equations. We can choose the μ-values in a variety of ways. Two representative methods are suggested here.

Specifying $\mu_1, \ldots, \mu_k$, $k \leq n - m$. The canonical form of the linear programming problem that the simplex method solves is:

Minimize the *goal function* $\mathbf{c}^T\mathbf{y}$ subject to the constraints

$$\mathbf{Ay} = \mathbf{b},$$

where the vector $\mathbf{y}$ is unknown, and $\mathbf{y} > \mathbf{0}$.

In this case, we specify k positive values of $\mathbf{M}_\mu$ and minimize the maximum of the $(n - k)$ unspecified values of $\mathbf{M}_\mu$ subject to the constraints $\mu_{k+1}, \ldots, \mu_n > 0$, and $c_1, \ldots, c_{n-m} > 0$. In other words, we have the following equations

$$\begin{aligned}
a_{1,1}c_1 + \cdots + a_{1,n-m}c_{n-m} &= \mu_1 \\
&\vdots \\
a_{k,1}c_1 + \cdots + a_{k,n-m}c_{n-m} &= \mu_k \\
a_{k+1,1}c_1 + \cdots + a_{k+1,n-m}c_{n-m} &\leq \varepsilon \\
&\vdots \\
a_{n,1}c_1 + \cdots + a_{n,n-m}c_{n-m} &\leq \varepsilon,
\end{aligned}$$

where $c_i \geq 0$, $\varepsilon > 0$, and we want to find $\mathbf{c}$ which minimises ε.

To convert the $n - k$ inequalities to equalities, we subtract ε from both sides of the equation and add *slack variables* $z_1, \ldots, z_{n-k}$ to give us the following $n - k$ equations

$$\begin{aligned}
a_{k+1,1}c_1 + \cdots + a_{k+1,n-m}c_{n-m} - \varepsilon + z_1 &= 0 \\
&\vdots \\
a_{n,1}c_1 + \cdots + a_{n,n-m}c_{n-m} - \varepsilon + z_{n-k} &= 0,
\end{aligned}$$

in addition to the first k equations. Now we have n equations with $2n - m - k$ unknown nonnegative quantities $(c_1, \ldots, c_{n-m}, z_1, \ldots, z_{n-k})$.

Let us label ε as c_0. By inspection, we see that the goal function to be minimised is c_0, subject to the constraints $\mathbf{A}'\mathbf{c}' = \mathbf{M}'_\mu$, where $\mathbf{c}'$ is a $(2n - m - k + 1)$-dimensional vector $\mathbf{M}'_\mu$ is a n-dimensional vector, and $\mathbf{A}'$ is an n by $2n - m - k + 1$ matrix; that is, we require to solve

$$\begin{pmatrix} 0 & & 0 & & \\ \vdots & & & \ddots & \\ 0 & \mathbf{A} & & & 0 \\ -1 & & 1 & & \\ \vdots & & & \ddots & \\ -1 & & & & 1 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-m} \\ z_1 \\ \vdots \\ z_{n-k} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_k \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{8}$$

and $c_i, z_i \geq 0$. This is in the canonical form for the simplex method.

Specifying $\mu_1, \ldots, \mu_n$. In this case, we specify all the values of $\mathbf{M}_\mu$. We indicate two possible options when solving for $\mathbf{c}$.

1. Minimise the mean-square error given by

$$\|\mathbf{Ac} - \mathbf{M}_\mu\|^2 = \sum_{i=1}^{n} (a_{i,1}c_1 + \cdots + a_{i,n-m}c_{n-m} - \mu_i)^2$$

subject to the constraints $\mathbf{M}_\mu > 0$, $\mathbf{c} > \mathbf{0}$.

This is a quadratic programming problem. However, this problem can be reformulated as a simplex method problem and can be solved using a variation of the traditional simplex method called Wolfe's method (Wolfe, 1959).

2. Minimise the largest absolute error c_0, given by

$$\max(|\varepsilon_1|, \ldots, |\varepsilon_n|)$$

where ε_i, the error in μ_i, is

$$\mu_i - (a_{i,1}c_1 + \cdots + a_{i,n-m}c_{n-m}), \quad i = 1, \ldots, n.$$

Our problem now is to minimize c_0 subject to c_0, $c_1, \ldots, c_{n-m} \geq 0$. To solve this problem,[1] we

[1] This is known as Chebyshev's Approximation (Franklin, 1980, p. 8).

note that we have n pairs of inequality contraints of the form

$$-c_0 + a_{i,1}c_1 + \cdots + a_{i,n-m}c_{n-m} \leq \mu_i,$$

$$-c_0 - a_{i,1}c_1 - \cdots - a_{i,n-m}c_{n-m} \leq -\mu_i.$$

The addition of slack variables puts the problem in canonical form.

2.2.4. *Characterisation of the Dual Spectral Scheme.* For simplicity, we consider algorithms employing the first linear programming approach outlined above. We have modified the initial basis for the nullspace of $\mathbf{U}$ using the results of the simplex method such that

$$\mathbf{W}^d = \mathbf{M} - \mathbf{Y}\mathbf{Y}^T,$$

where $\mathbf{M} = \mathbf{dg}[\mu_1, \mu_2, \ldots, \mu_n]$ with $\mu_1, \ldots, \mu_k > 0$ specified by us, and $0 < \mu_{k+1}, \ldots, \mu_n \leq \varepsilon < \min(\mu_1, \ldots, \mu_k)$, and $\mathbf{Y} = \mathbf{Xb}$ is a set of basis vectors for the left nullspace of $\mathbf{U}$. Since μ_i, $i = 1, \ldots, n$, are positive, we see that all the memories are strictly stable in the dual spectral scheme as long as the memories $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(m)}$ are linearly independent, and we are able to find the vector $\mathbf{c}$ in the system (8) through linear programming.

As asserted earlier, since $\mathbf{W}^d$ is a symmetric, zero-diagonal matrix, there exist Lyapunov functions for this scheme in both modes of operation. We have also conjectured that the attraction is directional in nature. The storage capacity of the dual spectral scheme of eqn (6) is directly $n - 1$. Specifically $n - 1$ is the number of memories for which we can still specify a left nullspace $\mathbf{X}$. (By Komlós' result (Komlós, 1967), we are guaranteed that almost all choices of n memories or fewer are linearly independent, so that for almost all choices of $n - 1$ memories there is an orthogonal subspace of dimension 1, while almost all choices of n memories span the space $\mathbf{R}^n$ and therefore the orthogonal subspace is of dimension 0.)

To find an n-dimensional vector under constraints, the simplex method iterates from one feasible solution to another until it finds an optimal feasible solution. The maximum number of iterations that the simplex method can go through to find an n-dimensional vector is $2^n - 1$.[2] However, it has been widely reported (Chvátal, 1983; Murty, 1983) that, in practice, the number of iterations is almost always between 1 to 3 times the number of constraints. Thus, for the case of specifying k values of $\mathbf{M}_\mu$, we would expect at the most $3n$ iterations. The computational complexity of each iteration is dependant on how the simplex method is implemented. For the revised simplex method, a good estimate of the average cost of each iteration in our scheme is $52n - 10m - 10k + 10$, while for the standard simplex method, a good estimate is $(2n^2 - mn - kn + n)/4$ (cf. Chvátal, 1983, p. 113). Thus, we estimate that the total cost of specifying k values of $\mathbf{M}_\mu$ is $O(n^2)$ (using the revised simplex method). The cost of finding a basis for the nullspace of $\mathbf{U}$ (through Gram–Schmidt orthogonalisation) includes finding $(\mathbf{U}^T\mathbf{U})^{-1}$ and two other matrix multiplications and is given by $mn^2 - (m^2n)/2 - m^3/2 + O(n^2)$. Finally, the cost of finding $\mathbf{W}^d$ from $\mathbf{c}$ and $\mathbf{X}$ is $n^3 - n^2m + O(n^2)$. So, we can say that on the average,

$$N^d = n^3 + \tfrac{1}{2}m^2n - mn^2 - m^3/2 + O(n^2),$$

where N^d is the number of elementary operations needed to compute $\mathbf{W}^d$.

There are a number of open questions involved with the dual spectral scheme arising from the nature of the construction of the $\mathbf{W}^d$ matrix. The number of directions k, that can be specified given a set of m memories and n neurons is of interest. It is obvious from the previous discussion about the dimensions of $\mathbf{A}$ and $\mathbf{c}$, that we can surely specify no more than $n - m$ directions. However, there is a possibility (albeit small) that there exist no feasible solutions for pathological cases where $k < n - m$. This is seen particularly when the number $n - m$ is very small. Another quantity we are interested in is the size of ε, the largest of the unspecified μ's, compared to the size of the specified μ's since we have conjectured that this will affect directional attraction.

While there exists little theory for the simplex method which will enable us to gauge these parameters, simulations show that ε is typically small compared to μ_i for the specified directions ($<0.5\mu_i$), and k is typically of the order of $n/4$ in the ranges simulated. We conjecture that this behaviour continues to hold for large n.

2.3. Composite Algorithms

In section 2.1 we saw ways of increasing the radii of attraction-spheres around memories. In section 2.2 we say ways of specifying increased attraction in certain directions around each of the memories. A natural extension of these schemes is to create a composite scheme with weight matrix $\mathbf{W}^c$ given by

$$\mathbf{W}^c = \mathbf{W}^s + \mathbf{W}^d.$$

Since $\mathbf{W}^c$ is a linear combination of $\mathbf{W}^s$ and $\mathbf{W}^d$, we would expect memories to be stable in the composite scheme for reasons described in the previous sections. The idea of the composite scheme is to specify both memory-specific attraction by specifying λ for each memory, and direction-specific attraction by specifying μ for the individual directions (Figure 3).

[2] This happens when the simplex method tests each vertex of the n-sided polyhedron that bounds the feasible region.

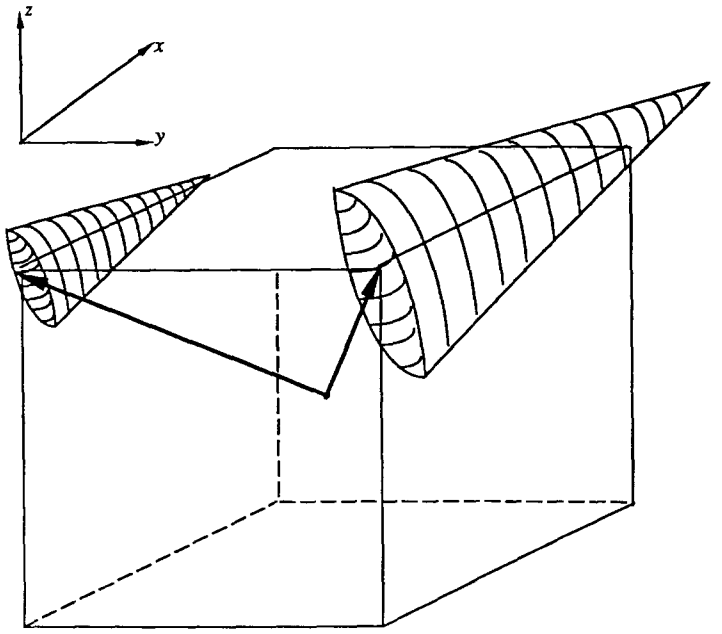

FIGURE 3. Schematic representation of the joint memory-specific and direction-specific attraction space for two memories in the composite scheme.

Here, the spectrum of $\mathbf{W}^s$ is no longer degenerate, and $\mathbf{W}^c$, consequently, is no longer symmetric. As the composite algorithm combines the memory-specific spectral algorithm, and the direction-specific dual spectral algorithm, it works effectively in shaping the attraction regions as desired. It should be noted that the relative values of the $\lambda^{(1)}, \ldots, \lambda^{(m)}$, compared to the $\mu_1, \ldots, \mu_n$, need to be considered in order not to lose the effects of one of the two parts of the composite scheme.

Note that the capacity of the composite scheme is $n - 1$. The algorithm complexity of the composite scheme is the sum of the complexities of the spectral and dual spectral schemes, except that we need not find $(\mathbf{U}^T\mathbf{U})^{-1}$ twice. Therefore the complexity N^c is given by $3n^3 + O(n^2)$ for $m \leq n$.

3. SIMULATIONS

Computer simulations were carried out to verify the behaviour of the various schemes. Systems with state vectors of 32 bits were considered in the simulations. The memories were chosen randomly with a binomial pseudo-random number generator with equiprobable values 1 and -1. For each size of memory set m that was investigated, simulations were carried out for each of the schemes, and the behaviour of the schemes was averaged out over between 20 and 100 trials, where over each trial a different random set of memories was generated. Error correction data were compiled at each trial by testing the convergence of randomly generated probes at increasing Hamming distance from a memory. Attraction radii were estimated by averaging the maximum error correction radius for each trial over the number of trials. The graphs included here were obtained from synchronous mode operations. However, we found that the schemes essentially behaved the same under an asynchronous mode of operation. The graphs show typical stability and attraction behaviour in each of the schemes. We can extract information on expected worst and best case behaviour for a set of random memories from these curves.

The behaviour of the outer product scheme is highlighted in Figures 4 and 5. As anticipated, the

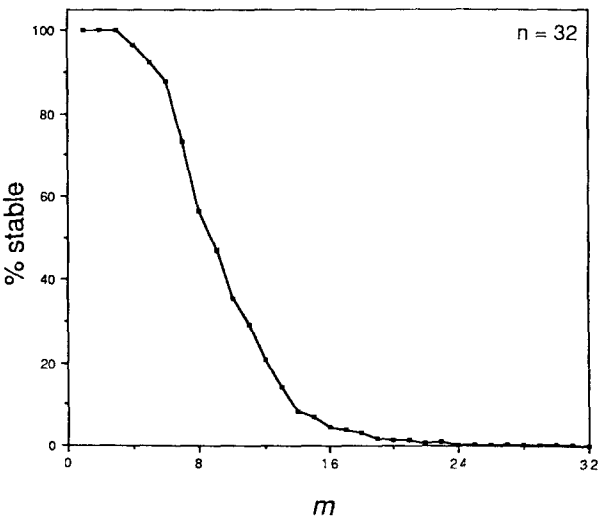

FIGURE 4. The percentage of stable memories plotted against the number of memories m in the outer product scheme when $n = 32$.

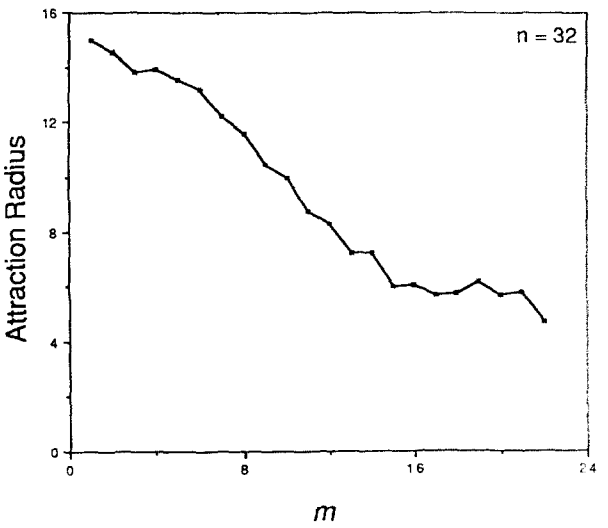

FIGURE 5. The average radius of attraction around a stable memory is plotted versus the number of memories for n = 32 in the outer-product scheme. The attraction radius is estimated by averaging the maximum Hamming distance of error-correction around a stable memory over several independent runs.

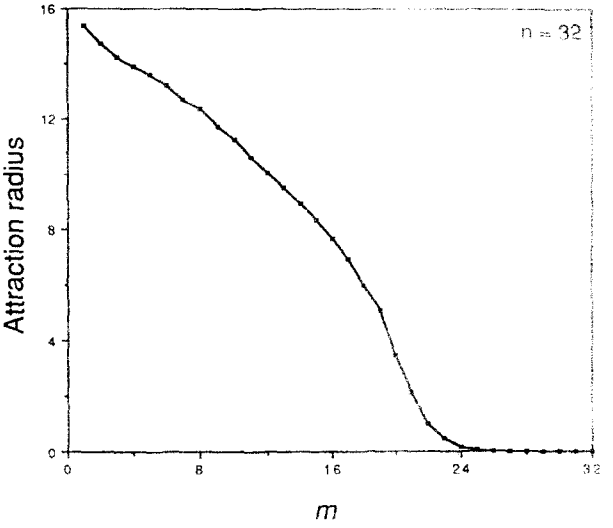

FIGURE 6. The attraction radius around a typical memory plotted as a function of the number of memories m, in the degenerate spectral scheme where all the eigenvalues are chosen equal to $\lambda = n = 32$. Estimates of the attraction radius for a given number of memories were again obtained by averaging the maximum distance of error-correction around a memory over several independent runs.

number of stable memories declines precipitously as m increases beyond a certain point (the static capacity) as seen in Figure 4. While n is quite small in these examples, the figures nonetheless are a precursor of the 0–1 behaviour which develops around the static capacity of $n/(4 \log n)$ for large n (Venkatesh, 1986; McEliece et al., 1987; Komlós & Paturi, 1988). Figure 5 shows the graceful degradation of the average Hamming radius of attraction around the memories as the number of stored memories increases. (We averaged the maximum attraction radius for each of the memories over several independent trials to obtain estimates of the average radius of attraction.) The analysis in McEliece et al. (1987) indicates that the attraction is neither memory- nor direction-specific, and that we obtain uniform Hamming balls of attraction around each memory with high probability for large n.

Simulations highlighting the behaviour of the spectral scheme as a viable algorithm for associative memory are presented in Figures 6 and 7. The average Hamming radius of attraction again degrades gracefully as the number of memories increases, as illustrated in Figure 6, where the degenerate spectral algorithm exhibits uniform balls of attraction around the memories. (The static capacity here is clearly n as outlined before and verified in our simulation.) As can be seen, the dynamical behaviour of the spectral scheme is qualitatively similar to the outer product scheme, but somewhat better over all ranges.

Investigations into attraction dynamics in the spectral scheme when there is a large deviation in eigenvalue size confirm theoretical predictions that the sizes of the attraction basins are memory-specific and increase with increase in the eigenvalue size of the corresponding memory. These trends are exemplified in the typical plot of Figure 7 where half the eigenvalues are fixed arbitrarily at n, and the other half of the eigenvalues are fixed at a fraction of n. The plots show the relative sizes of the Hamming balls of attraction for memories with large eigenvalue as compared to memories with small eigenvalue, as a function of the ratio of the two eigenvalues. The results are similar for other values of m in the range of interest (i.e., values of m for which there is significant attraction: the attraction radii around the memories is proportional to corresponding eigenvalue size).

The feasibility of forming the dual spectral matrix $\mathbf{W}^d$, using the simplex method when $\mu_1, \ldots, \mu_k$ are specified is confirmed in Figures 8 and 9. The success rate (the percentage of trials when the simplex method returns a feasible solution with $\varepsilon < \min(\mu_1, \ldots, \mu_n)$) is plotted in Figure 8 against the number of memories m, averaged over various choices of k. In Figure 9, the success rate is plotted as a function of the number of specified directions k, with m as a parameter. Note that the success rate is almost 100% when k is small, and drops gradually with failures occurring most often when k approaches $n - m$ (Figure 9). Figure 10 exhibits plots of average ε versus k for various m. As can be seen, ε increases with increasing k and increasing m. Exhaustive simulations indicate that the values of ε obtained by the simplex algorithm for $n = 16$ (fixed m, k) are approximately twice those for $n = 32$. Since the

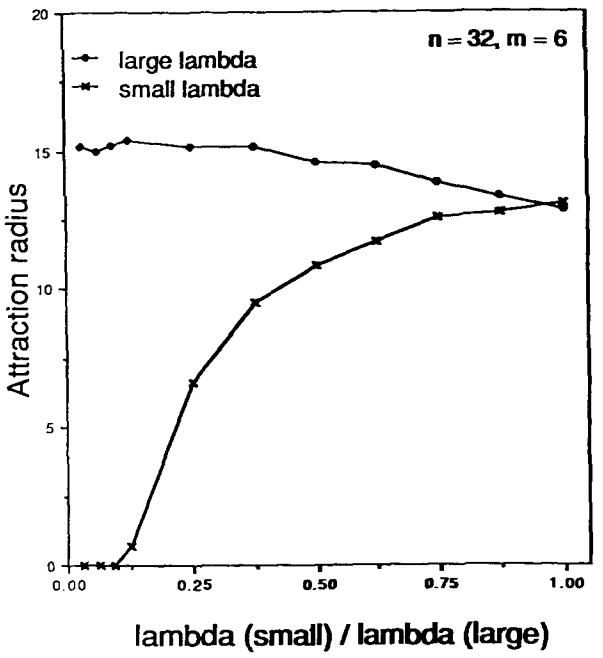

FIGURE 7. Demonstration of memory-specific attraction in the spectral scheme for $n = 32$ and $m = 6$. The memories were divided into two equal sized groups, one group with eigenvalue λ(large) = n, and the other group with eigenvalue λ(small) varying as a fraction of n. The respective attraction radii of the λ(large) memories and the λ(small) memories are plotted as the ratio λ(small)/λ(large) is increased from zero to one.

dynamic attraction behaviour of the dual spectral scheme is dependent on the size of ε, these curves are crude indicators of the limits on m and k in the dual spectral scheme.

Investigations into the attraction dynamics of the

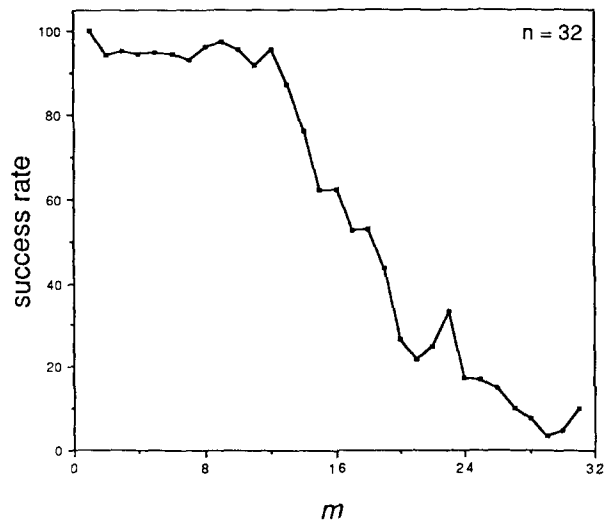

FIGURE 8. The percentage of trials when the simplex method returns a feasible solution (the success rate) in forming the dual spectral matrix W^d, averaged over various choices of k, the number of specified directional values, $\mu_1, \ldots, \mu_k$, plotted as a function of the number of memories m, when $n = 32$.

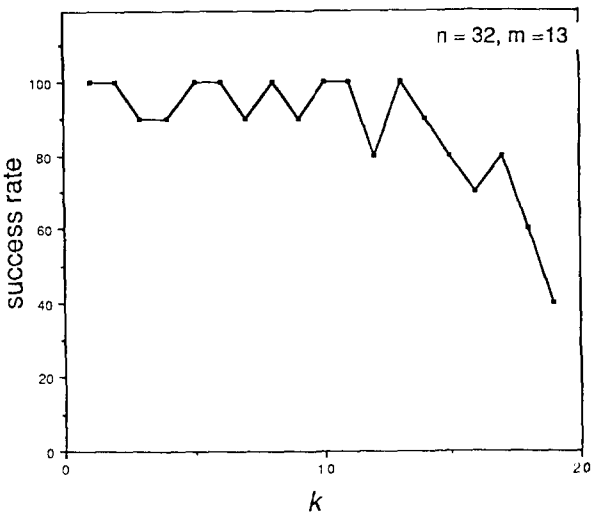

FIGURE 9. The percentage of trials when the simplex method returns a feasible solution for the dual spectral scheme (the success rate) plotted as a function of the number of specified directions k, for a choice of $n = 32$, $m = 13$. (Here k denotes the number of directional values, $\mu_1, \ldots, \mu_k$, specified in the algorithm.)

dual spectral scheme verify the analytical predictions of its performance as an associative memory. We will use the measure of attraction in a particular direction x for a particular memory to be the average Hamming radius from which state vectors converge to that memory when bit x is kept flipped. (Specifically, if mu_x is large, then inputs with bit x opposite in sign to a memory will be unlikely to converge to the memory, and conversely if μ_x is small. Equivalently, if bit

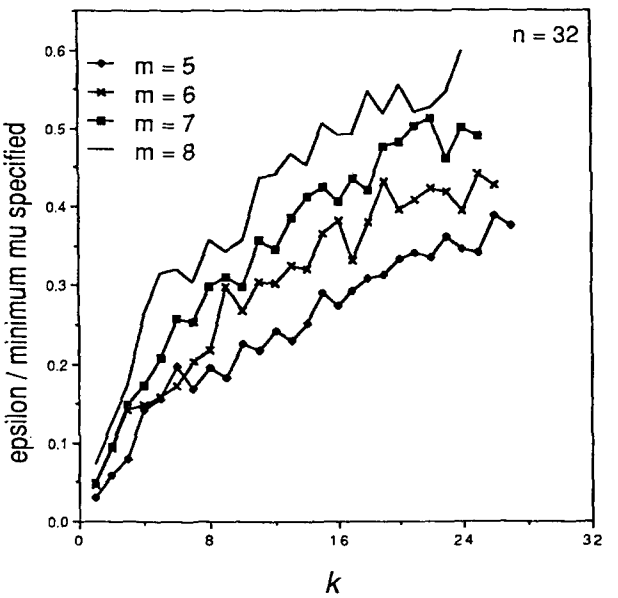

FIGURE 10. The ratio of the largest value ϵ, of the unspecified directional parameters, $\mu_{k+1}, \ldots, \mu_n$, to the smallest of the specified directional parameters $\mu_{\min} = \min\{\mu_1, \ldots, \mu_k\}$, plotted versus k, the number of specified directional parameters with m as a parameter for $n = 32$.

x of the input vector is constrained to be correctly matched to the corresponding bit of the memory, then the algorithm will tend to correct for rather large distortions in the other components if μ_x is large, and conversely if μ_x is small.) Figure 11 exhibits plots of average attraction in both the specified (important) and the unspecified (unimportant) directions, where the component of the input in the direction being investigated was initially kept flipped. Here, the attraction characteristics have been averaged over all the memories for the two cases: (1) the specified directions (corresponding to large values of μ), and (2) the unspecified directions (corresponding to small values of μ). As can be seen, there exists a consistent difference in attraction in the large μ and small μ directions when k is small, with a merging of the attraction capabilities for larger k.

The simulations indicate that we do have the capability of separately achieving memory-specific and direction-specific attraction. Investigations into the composite scheme indicate that attraction basins can indeed be shaped over a wide range. Varying the values of the specified μ_i's, and large eigenvalues (λ_{lg}) and small eigenvalues (λ_{sm}) lead to attraction basins that range from being purely directional to completely "spherical" around memories. A sample case where $\lambda_{sm} = 1$, $\lambda_{lg} = 3$, and $\mu_{lg} = 6$ (which gives us $\varepsilon \leq 3$ for moderate values of k and m) is shown in Figures 12 and 13. Figure 12 exhibits plots of mem-

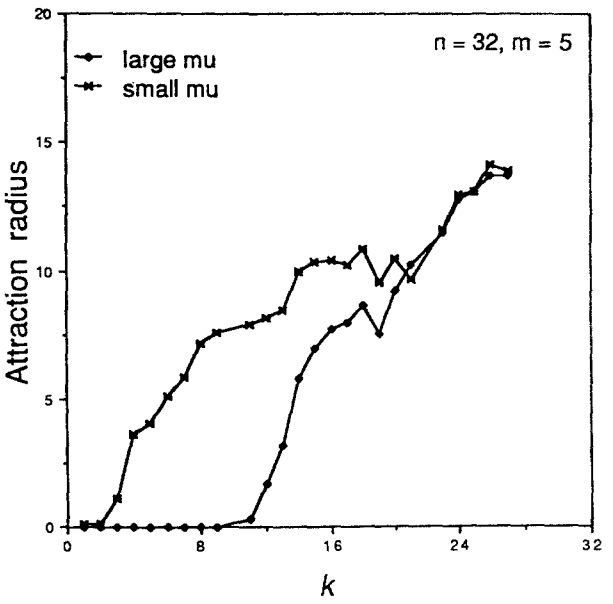

FIGURE 11. Demonstration of direction-specific attraction in the dual spectral scheme for $n = 32$ and $m = 5$. Curves of attraction radii versus the number of specified directional parameters k, are shown for two different directions—a specified (large μ) direction and an unspecified (small μ) direction. Attraction data for a given direction were generated by investigating probe vectors at various Hamming distances from a memory with the component of the probe in the direction being investigated being chosen to be opposite in sign to the corresponding component of the memory. (Flipping a bit in an important (large μ) direction would reduce the attraction to the memory compared to an unimportant (small μ) direction.)

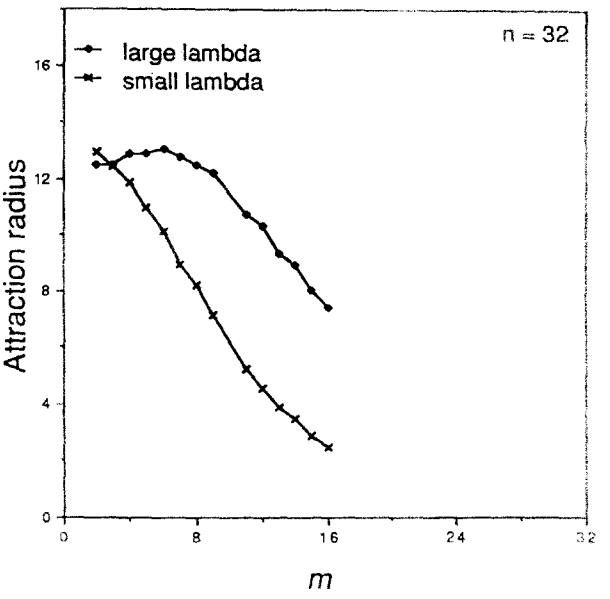

FIGURE 12. Demonstration of memory-specific attraction in the composite scheme for $n = 32$. The memories are divided into two groups, one group corresponding to a "large" eigenvalue $\lambda_{lg} = 3$, and the other group corresponding to a "small" eigenvalue $\lambda_{sm} = 1$. Attraction radii for a memory are plotted as a function of the number of memories m for the two cases of the memory corresponding to eigenvalues $\lambda_{lg} = 3$ and $\lambda_{sm} = 1$.

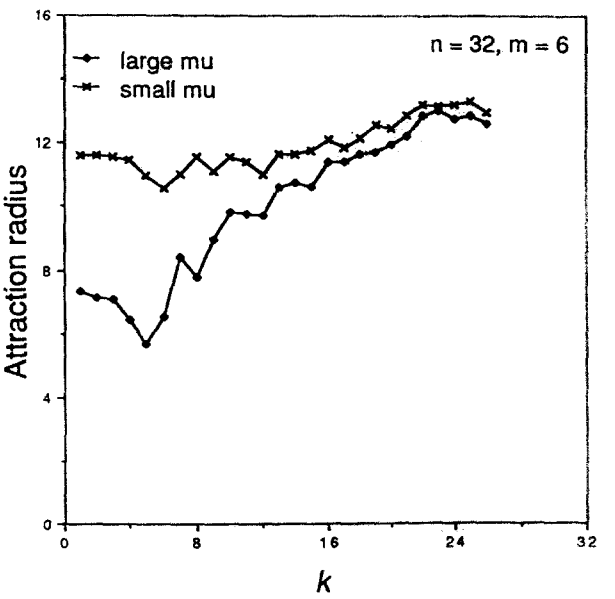

FIGURE 13. Demonstration of direction-specific attraction in the composite scheme for $n = 32$ and $m = 6$. Directional attraction parameters are specified to be all equal to $\mu_{lg} = 6$, while the largest of the unspecified directional parameters is kept below $\epsilon = 3$. Attraction radii are plotted in the large μ (specified) and small μ (unspecified) directions as a function of k, the number of specified directions.

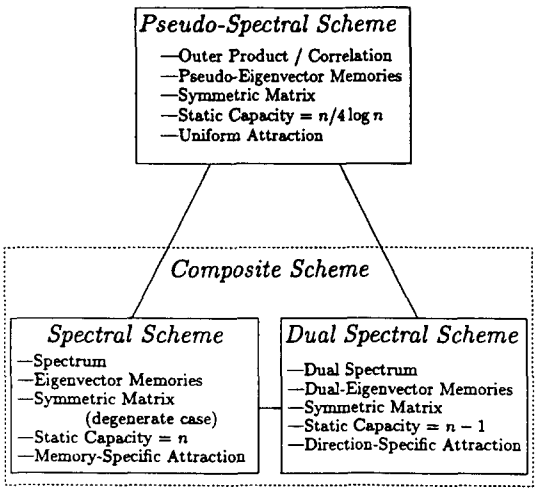

FIGURE 14. An overview of the features of the family of spectral algorithms—the outer product (pseudo-spectral) algorithm, the spectral algorithm, the dual spectral algorithm, and the composite algorithm.

ory-specific attraction against the number of memories. As can be seen, there is a superiority of between 6 to 8 Hamming bits of attraction for memories with large eigenvalues as opposed to memories with small eigenvalues. (For $n = 16$ we obtain a superiority of between 2 to 3 Hamming bits of attraction for the same choice of maximum and minimum eigenvalues.) Direction-specific attraction is mapped in Figure 13. As seen, we obtain a direction-specificity of about 4 bits in attraction capability when comparing the strong and weak directions. (For $n = 16$ we perceive a 2 to 3 bit difference in attraction capability between specified and unspecified directions for small values of k. When the number of memories m is very small, however, only marginal direction-specificity is displayed.) We stress once again that by increasing the value of the specified μ's, we increase direction-specific attraction at the expense of memory-specific attraction.

Figure 14 summarises the main features of the three algorithms, and highlights their relationship with the spectral algorithm: in particular, the pseudo-spectral nature of the outer-product algorithm, and the dual spectral nature of the dual spectral algorithm is emphasised.

REFERENCES

Amari, S. (1977). Neural theory of association and concept formation. *Biological Cybernetics*, **26**, 175–185.

Chvátal, V. (1983). *Linear programming*. New York: W. H. Freeman.

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Transactions on Electronic Computers*, **EC-14**, 326–334.

Franklin, J. (1980). *Methods of mathematical economics*. New York: Springer-Verlag.

Goles, E., & Vichniac, G. Y. (1986). Lyapunov function for parallel neural networks. In J. Denker (Ed.), *Neural Networks for Computing*. AIP Conference Proceedings, **151**, 165–181.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA*, **79**, 2554–2558.

Kohonen, T. (1977). *Associative memory: A system-theoretic approach*. Berlin Heidelberg: Springer-Verlag.

Komlós, J. (1967). On the determinant of (0, 1) matrices. *Studia Scientarum Mathematicarum Hungarica*, **2**, 7–21.

Komlós, J., & Paturi, R. (1988). Convergence results in an associative memory model. *Neural Networks*, **1**, 239–250.

Maruani, A. D., Chevallier, R. C., & Sirat, G. (1987). Information retrieval in neural networks. I. Eigenproblems in neural networks. *Rev. Phys. Appl.*, **22**, 1321–1325.

McCulloch, W. W., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin Mathematical Biophysics*, **5**, 115–133.

McEliece, R. J., Posner, E. C., Rodemich, E. R., & Venkatesh, S. S. (1987). The capacity of the Hopfield associative memory. *IEEE Transactions on Information Theory*, **IT-333**, 461–482.

Murty, K. G. (1983). *Linear programming*. New York: Wiley.

Nakano, K. (1972). Associatron—A model of associative memory. *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-2**, 380–388.

Peretto, P., & Niez, J. J. (1986). Long term memory storage capacity of multiconnected neural networks. *Biological Cybernetics*, **54**, 53–63.

Personnaz, L., Guyon, I., & Dreyfus, G. (1985). Information storage and retrieval in spin-glass like neural networks. *Journal Physique Lettres*, **46**, L359–L365.

Psaltis, D., & Venkatesh, S. S. (1989). Information storage in fully connected networks. In Y. C. Lee (Ed.), *Evolution, learning, and cognition* (pp. 51–89). Teaneck, New Jersey: World Scientific.

Strang, G. (1980). *Linear algebra and its applications*. New York: Academic Press.

Venkatesh, S. S., & Psaltis, D. (1985). Efficient strategies for information storage and retrieval in associative neural nets. *Workshop on Neural Networks for Computing*, Santa Barbara, California.

Venkatesh, S. S. (1986a). Epsilon capacity of neural networks. In J. Denker (Ed.), *Neural networks for computing*. AIP Conference Proceedings, **151**, 440–445.

Venkatesh, S. S. (1986b). *Linear maps with point rules: Applications to pattern classification and associative memory*. Ph.D. thesis, California Institute of Technology.

Venkatesh, S. S., & Psaltis, D. (1989). Linear and logarithmic capacities in associative neural networks. *IEEE Transactions on Information Theory*, IT-35, 558–568.

Wolfe, P. (1959). The simplex method for quadratic programming. *Econometrica*, **27**, 282–298.