JOURNAL OF COMPLEXITY 7, 316–337 (1991)

Programmed Interactions in Higher-Order Neural Networks: Maximal Capacity*

SANTOSH S. VENKATESH†

Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia, Pennsylvania 19104

AND

PIERRE BALDI‡

Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109

Received February 15, 1989

The focus of the paper is the estimation of the maximum number of states that can be made stable in higher-order extensions of neural network models. Each higher-order neuron in a network of n elements is modeled as a polynomial threshold element of degree d. It is shown that regardless of the manner of operation, or the algorithm used, the storage capacity of the higher-order network is of the order of one bit per interaction weight. In particular, the maximal (algorithm independent) storage capacity realizable in a recurrent network of n higher-order neurons of degree d is of the order of $n^d/d!$. A generalization of a spectral algorithm for information storage is introduced and arguments adducing near optimal capacity for the algorithm are presented. © 1991 Academic Press, Inc.

1. INTRODUCTION

A formal neuron (after McCulloch and Pitts, 1943) is defined as a linear threshold element which accepts n inputs and computes a binary output based on the sign of a linear form of the inputs. When n such elements are

* Presented in part at the IEEE Conference on Neural Information Processing Systems, Denver, Colorado, November, 1987, and at the IEEE International Symposium on Information Theory, Kobe, Japan, June, 1988.

† Corresponding author.

‡ Also Division of Biology, California Institute of Technology, Pasadena, CA 91125.

0885-064X/91 $3.00

interconnected with the output of each neuron serving as input to all the neurons in the network, a closed feedback system results with dynamics described by trajectories on the vertices of the n-cube. Each vertex defines a possible state of the recurrent network, and we identify the vector of neural outputs as the (instantaneous) state of the system. The fixed points (or stable states) of such recurrent networks are of importance in their computational characterization; in particular, we are interested in the following question: What is the maximum number of arbitrarily specified vertices that can be made stable in a recurrent neural network by suitable selection of neural interconnectivity?

In this paper we focus on recurrent networks where the computational elements are higher-order extensions of the basic linear threshold neural model. Each higher-order neuron is a *polynomial threshold element* of a given degree d. If, in a recurrent network of n higher-order neurons, the current outputs (states) of the neurons are $u_1, \ldots, u_n \in \{-1,1\}$, then an update, u'_{i_1}, of the state of the i_1th neuron is given by the sign of an algebraic form

$$u'_{i_1} = \operatorname{sgn}\left(\sum_{1 \le i_2 \le \cdots \le i_{d+1} \le n} w_{i_1 i_2 \cdots i_{d+1}} u_{i_2} \cdots u_{i_{d+1}}\right). \tag{1}$$

The number of degrees of freedom in choosing the interaction coefficients (or weights) $w_{i_1 i_2 \cdots i_{d+1}}$ is increased to n^{d+1} from the n^2 weights for the case of linear interactions. The added degrees of freedom in the interaction coefficients can potentially result in enhanced flexibility and programming capability over the linear case as has been noted independently by several authors (Lee *et al.*, 1986; Psaltis and Park, 1986; Baldi and Venkatesh, 1987, 1988).

We rigorously estimate the storage capacity of recurrent higher-order neural networks: specifically, we calculate the maximum number of arbitrarily specified vectors that can be made stable in a recurrent network of n polynomial threshold units of degree d.[1] All our results point in the following direction.

Regardless of the manner of operation, or the algorithm utilized, the storage capacity of a higher-order network of degree d is of the order of 1 *memory bit per interaction coefficient. And in particular:*

- *The storage capacity of the outer-product algorithm generalized to networks of degree d is of the order of* $n^d/\log n$ *memories (with constants depending on the variant employed);*

[1] Cases where networks have random interaction coefficients (instead of the programmed scenario here) lead to entirely different computational issues. We deal with these in a concurrent paper (Venkatesh and Baldi, 1989).

• *The maximal (algorithm independent) storage capacity realizable in a higher-order neural network of degree d is of the order of $n^d/d!$;*

• *Near optimal storage capacities of the order of $n^d/d!$ memories can be obtained by variants of the spectral algorithm.*

In this paper we set up the basic definitions in Section 2, construct a spectral based algorithm with near optimal capacity in Section 3, and rigorously estimate the maximal (algorithm independent) capacity of a network of given degree in Section 4. In a concurrent paper we include the capacity calculations for the outer-product algorithm generalized to degree d (Venkatesh and Baldi, 1991).

Notation. Let $\{x_n\}$ and $\{y_n\}$ be positive sequences. We use the following standard asymptotic notation:

1. $x_n = O(y_n)$ if there is a positive constant L such that $x_n/y_n \leq L$ for all n;
2. $x_n \sim y_n$ if $x_n/y_n \to 1$ as $n \to \infty$;
3. $x_n = o(y_n)$ if $x_n/y_n \to 0$ as $n \to \infty$.

By *almost all* we mean all but an asymptotically negligible subset: specifically, if A_n denotes a sequence of finite sets, and $\mathcal{P}$ is some attribute, we say that almost all elements of A_n exhibit $\mathcal{P}$ if the subsets $B_n \subseteq A_n$ for which $\mathcal{P}$ holds are such that $|B_n| \sim |A_n|$ as $n \to \infty$. We denote by $\mathbb{B}$ the set $\{-1, 1\}$, and by $[n]$ the set of indices $\{1, 2, \ldots, n\}$ for any positive integer n. Finally, by an *ordered multiset* we mean an ordered collection of elements where repetition is allowed.

2. Higher-Order Neural Networks

2.1. *Polynomial Threshold Units*

We consider recurrent networks of polynomial threshold units each of which yields an instantaneous state of -1 or $+1$. More formally, for positive integers n and d, let $\mathcal{I}_d$ be the set of ordered multisets of cardinality d of the set $[n]$. Clearly $|\mathcal{I}_d| = n^d$. For any subset I of $[n]$, and for every $\mathbf{u} = (u_1\, u_2 \cdots u_n) \in \mathbb{B}^n$, set $u_I = \Pi_{i \in I}\, u_i$.

Definition 2.1. A *fully interconnected higher-order neural network of degree d* is characterized by a set of n^{d+1} real *weights* $w_{(i,I)}$ indexed by the ordered pair (i,I) with $i \in [n]$ and $I \in \mathcal{I}_d$, and a real *margin of operation* $\mathcal{B} \geq 0$. The network dynamics are described by trajectories in a state space of binary n-tuples, $\mathbb{B}^n$: for any state $\mathbf{u} \in \mathbb{B}^n$ on a trajectory, a component update $u_i \mapsto u_i'$ is permissible iff

$$u_i' = \begin{cases} -1 & \text{if } \sum_{I \in \mathcal{I}_d} w_{(i,I)} u_I < -\mathcal{B} \\ -u_i & \text{if } -\mathcal{B} \le \sum_{I \in \mathcal{I}_d} w_{(i,I)} u_I \le \mathcal{B} \\ 1 & \text{if } \sum_{I \in \mathcal{I}_d} w_{(i,I)} u_I > \mathcal{B}. \end{cases} \tag{2}$$

The evolution may be *synchronous* with all components of $\mathbf{u}$ being updated according to the rule (2) at each epoch, or *asynchronous* with at most one component being updated per epoch according to Eq. (2).

The network is said to be *symmetric* if $w_{(i,I)} = w_{(j,J)}$ whenever the $(d+1)$-tuples of indices (i, I) and (j, J) are permutations of each other. The network is said to be *zero-diagonal* if $w_{(i,I)} = 0$ whenever any index repeats in (i, I).

Let $\hat{\mathcal{I}}_d$ denote the set of all subsets of d elements from $[n]$; $|\hat{\mathcal{I}}_d| = \binom{n}{d}$. Combining all redundant terms in Eq. (2), for symmetric, zero-diagonal networks a component update $u_i \mapsto u_i'$ is permissible iff

$$u_i' = \begin{cases} -1 & \text{if } \sum_{I \in \hat{\mathcal{I}}_d : i \notin I} w_{(i,I)} u_I < -\mathcal{B} \\ -u_i & \text{if } -\mathcal{B} \le \sum_{I \in \hat{\mathcal{I}}_d : i \notin I} w_{(i,I)} u_I \le \mathcal{B} \\ 1 & \text{if } \sum_{I \in \hat{\mathcal{I}}_d : i \notin I} w_{(i,I)} u_I > \mathcal{B}. \end{cases} \tag{3}$$

(If the network is symmetric and zero-diagonal then, for each nonzero coefficient $w_{(i,I)}$—i.e., coefficients $w_{(i,I)}$ for which no index repeats in (i, I)—the term $w_{(i,I)} u_I$ occurs $d!$ times in the sum $\sum_{I \in \mathcal{I}_d} w_{(i,I)} u_I$. Hence, $\sum_{I \in \mathcal{I}_d} w_{(i,I)} u_I = d! \sum_{I \in \hat{\mathcal{I}}_d : i \notin I} w_{(i,I)} u_I$. The constant scale factor $d!$ is removed in Eq. (3) as this is just equivalent to scaling the margin.)

The choice of margin of operation essentially specifies the "strength" of the desired interaction. A choice of margin $\mathcal{B} = 0$ leads to standard threshold operation. For a choice of nonzero margin of operation, a bit, u_i, retains its sign if and only if the corresponding weighted sum multiplied by u_i exceeds $\mathcal{B}$; otherwise its sign is reversed.

These networks are seen to be natural generalizations to higher-order of the familiar case of *linear threshold networks* ($d = 1$). While networks of polynomial threshold units require more computationally powerful units than linear threshold functions, each polynomial threshold element (subscribing to rule (2) or to rule (3)) can be replaced by a small, equivalent network of linear threshold units. To see this note that it suffices to be able to realize each individual product of components, $u_I = \prod_{j=1}^{d} u_{i_j}$, for each choice of $I = (i_1, i_2, \ldots, i_d) \in \mathcal{I}_d$, as the results of all these computations can be combined with a single linear threshold gate to realize the desired output. Now, for each $I \in \mathcal{I}_d$, realizing the product of components u_I is equivalent to checking the parity of the d bits u_{i_1}, $u_{i_2}, \ldots, u_{i_d}$ in the product. It suffices, hence, to show that parity can be computed by small circuits of linear threshold units. But this, in fact, is a special case of a more general known result that any *symmetric Boolean*

function—i.e., functions which are invariant under any permutation of the inputs, parity being an example—can be computed by small circuits of linear threshold elements. For completeness, we sketch a short proof of this result below.

PROPOSITION 2.2. *Any symmetric Boolean function on d variables can be computed by a linear threshold circuit of depth two and linear size; in particular, d threshold elements in the first layer and a single output threshold element in the second layer are always sufficient.*

Proof. The proof is constructive. Array the 2^d possible inputs of ± 1 d-tuples in $(d + 1)$ rows with the elements in each row being permutations (i.e., all d-tuples in a row have the same number of $+1$'s), the lowest row containing the single d-tuple which has no $+1$'s, and with the number of $+1$'s increasing monotonically with the rows to the final $(d + 1)$th row which contains the single d-tuple whose components are all $+1$. Any symmetric Boolean function clearly assumes the same value for all elements (Boolean d-tuples) in a row. Hence, for any given symmetric function, contiguous rows where the function assumes the value $+1$ form *bands* which are separated by contiguous rows where the function assumes the value -1. This is illustrated schematically in Fig. 1a. Now assume there are b bands where the function assumes the value $+1$. (There are at most $d/2$ such bands—the worst case occurring for the parity function.) The function can now be computed by a circuit with $2b$ linear threshold elements in the first layer and a single linear threshold element in the second layer as illustrated in Fig. 1b. (Each linear threshold unit produces a $+1$ if the weighted sum of all its inputs exceeds its threshold, and produces a -1 otherwise.) ■

2.2. *Capacity*

As in any dynamical system, the fixed points are important in the characterization of the system dynamics.

DEFINITION 2.3. Let $\mathcal{B} \geq 0$ be fixed. A state $\mathbf{u} \in \mathbb{B}^n$ of a fully interconnected network is said to be $\mathcal{B}$-*stable* iff

$$u_i \sum_{I \in \$_d} w_{(i,I)} u_I > \mathcal{B}, \qquad i = 1, \ldots, n.$$

Likewise, a state $\mathbf{u} \in \mathbb{B}^n$ of a zero-diagonal network is said to be $\mathcal{B}$-*stable* iff

$$u_i \sum_{I \in \$_d; i \in I} w_{(i,I)} u_I > \mathcal{B}, \qquad i = 1, \ldots, n.$$

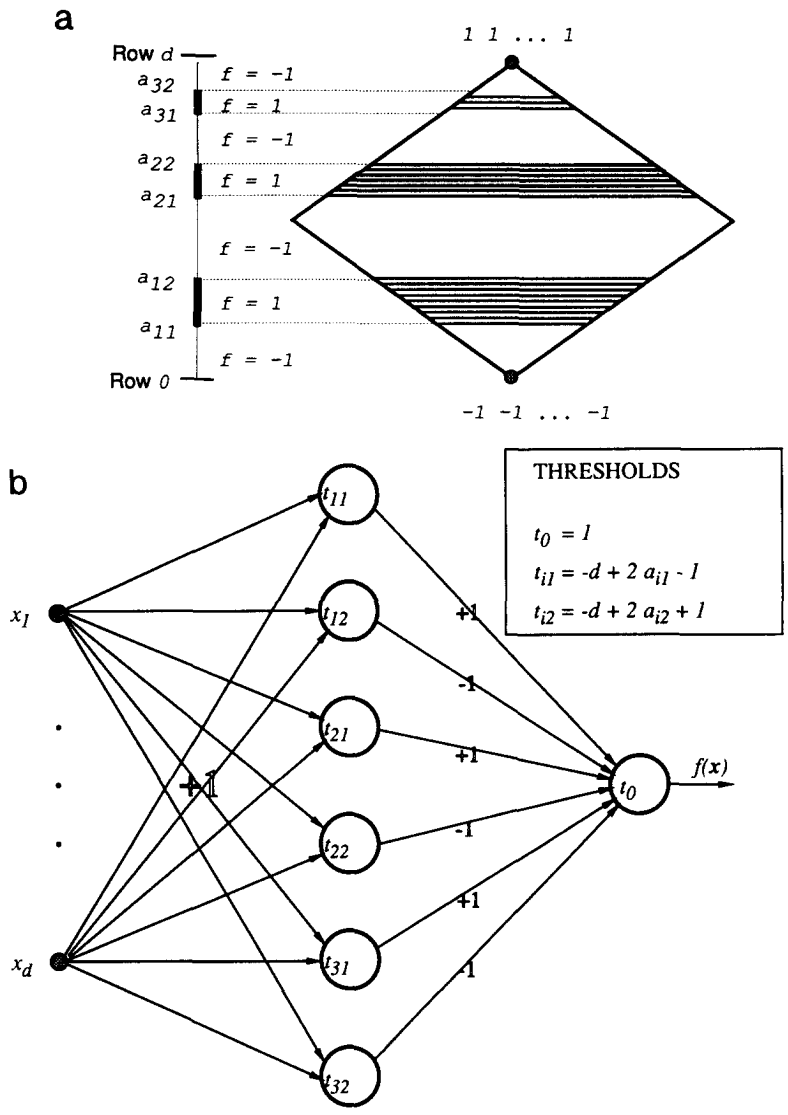

FIG. 1. (a) A symmetric Boolean function f of d inputs. (b) A realization of the symmetric Boolean function f with a linear number (in d) of linear threshold elements arrayed in a depth 2 circuit.

It is easy to see that $\mathcal{B}$-stable states are fixed points of the higher-order network with evolution under a margin $\mathcal{B}$.

The fixed points of the network take on particular significance when the network interconnections are symmetric. In this case, under suitable modes of operation, Liapunov functions can be shown for the system

(Hopfield, 1982; Goles and Vichniac, 1986; Maxwell *et al.*, 1986; Venkatesh and Baldi, 1989). In particular, each fixed point exhibits an *attraction basin;* trajectories passing through states in the attraction basin of a fixed point ultimately converge to the fixed point. This geometric picture is particularly persuasive in associative memory applications; if, by appropriate choice of weights, data is stored as fixed points of the network, then the network functions as an error-correction mechanism and identifies states sufficiently similar to a stored datum with the datum.

In this paper we do not insist on symmetry in the choice of weights. We refer to the data to be stored as *memories*. By an *algorithm* for storing memories we mean a prescription for generating the interaction weights of a higher-order network of degree d as a function of any given set of memories. We investigate the maximum number of arbitrarily specified memories that can be made fixed in the network by an algorithm; this is a measure of the *capacity* of the algorithm to store data.

Let $\mathbf{u}^1, \ldots, \mathbf{u}^m \in \mathbb{B}^n$ be an m-set of memories to be stored in a higher-order network of degree d. We assume that the memories are chosen randomly from the probability space of an unending series of symmetric Bernoulli trials: specifically, the memory components, u_i^α, $i \in [n]$, $\alpha \in [m]$, are i.i.d. random variables with

$$\mathbf{P}\{u_i^\alpha = -1\} = \mathbf{P}\{u_i^\alpha = +1\} = \tfrac{1}{2}.$$

In the following we assume that the network architecture is specified to be a higher-order network of degree d operating under a margin $\mathcal{B}$.

DEFINITION 2.4. We say that $\underline{C_n}$ is a *lower capacity function* (or simply, *lower capacity*) for an algorithm if for every $0 < \lambda < 1$, and $m \leq (1 - \lambda)\underline{C_n}$, the probability that all the memories are fixed points of the network generated by the algorithm tends to one as $n \to \infty$.

Likewise, $\underline{C_n}$ is a *maximal lower capacity* if for every $0 < \lambda < 1$, and $m \leq (1 - \lambda)\underline{C_n}$, the probability that there is some network in which all the memories are $\mathcal{B}$-stable approaches one as $n \to \infty$.

DEFINITION 2.5. We say that $\overline{C_n}$ is an *upper capacity function* (or simply, *upper capacity*) for an algorithm if for every $0 < \lambda < 1$, and $m \geq (1 + \lambda)\overline{C_n}$, the probability that at least one of the memories is not a fixed point of the network generated by the algorithm tends to one as $n \to \infty$.

Likewise, $\overline{C_n}$ is a *maximal upper capacity* if for every $0 < \lambda < 1$, and $m \geq (1 + \lambda)\overline{C_n}$, the probability that there is a network in which all the memories are $\mathcal{B}$-stable approaches zero as $n \to \infty$.

Remarks. The first definition yields an underestimate of algorithm/network capability, while the second definition gives an overestimate. Note that the definitions of maximal capacity are algorithm independent, and bound any algorithmic capacity from above. It is clear that both lower

and upper capacities always exist, and are not unique. What is more, there does not exist a largest lower capacity or a smallest upper capacity as the following proposition indicates. The proof is an immediate consequence of the definitions.

PROPOSITION 2.6. *(a) If $\underline{C_n}$ is a lower capacity, then so is $\underline{C_n}[1 \pm o(1)]$.*
(b) If $\overline{C_n}$ is an upper capacity, then so is $\overline{C_n}[1 \pm o(1)]$.

We combine the lower and upper estimates of capacity to obtain the following:

DEFINITION 2.7. C_n is a *capacity function* (or simply, *capacity*) for an algorithm iff it is both a lower and an upper capacity for the algorithm; it is a *maximal capacity* iff it is both a maximal lower and a maximal upper capacity.

Remarks. Capacity follows a 0–1 law. The probabilistic setup we espouse requires almost all sequences of memories within capacity to be storable as fixed points within the network. Capacity, hence, reflects typical behavior.[2] Figures 2a and 2b indicate the threshold behavior of capacity.

Unlike lower and upper capacity functions, capacity functions are not guaranteed to exist. If a capacity function exists, however, then it is not unique.

PROPOSITION 2.8. *If C_n is a capacity function, then so is $C_n[1 \pm o(1)]$; conversely, if C_n and C_n' are two capacity functions, then $C_n \sim C_n'$.*

Proof. The first part follows trivially because C_n is both a lower and an upper capacity. To prove the converse, let C_n and C_n' be any two capacity functions. Without loss of generality, let $C_n' = [1 + \alpha_n]C_n$. We must prove that $|\alpha_n| = o(1)$.

Let p denote the probability that all the memories are fixed points of the network. Fix $\lambda, \lambda' \in (0,1)$. For $m \leq (1 - \lambda')C_n' = (1 - \lambda')(1 + \alpha_n)C_n$, we have $p \to 1$ as $n \to \infty$. Further, for $m \geq (1 + \lambda)C_n$, we have $p \to 0$ as $n \to \infty$. Hence, for every choice of scalars $\lambda, \lambda' \in (0,1)$, we require that

$$1 + \alpha_n < \frac{1 + \lambda}{1 - \lambda'}$$

for large enough n. It hence follows that $|\alpha_n| = o(1)$. ■

[2] The definitions of capacity developed in this paper subsume within them most common notions of capacity, and can be easily extended in various ways to reflect properties of memories other than mere stability. For other variants, cf. Cover (1965), Vapnik (1982), Abu-Mostafa and St. Jacques (1985), Venkatesh (1986), Baldi (1988).

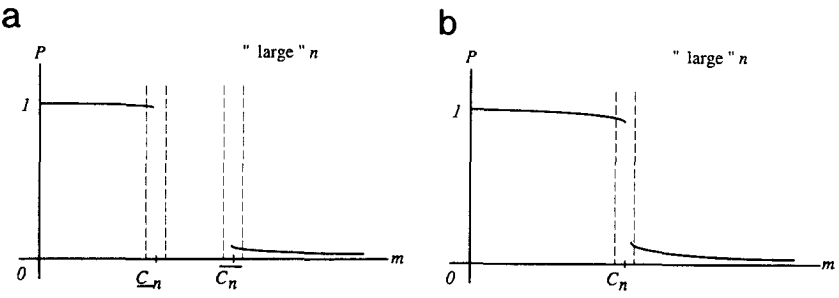

FIG. 2. (a) Lower and upper capacity functions; P denotes the probability that each of m randomly chosen memories is a fixed point of the network. (b) The 0–1 behavior of capacity.

Thus, if capacity functions do exist, they are not very different from each other asymptotically. Define the equivalence class $\mathscr{C}$ of (lower/upper) capacities by $C_n, C_n' \in \mathscr{C} \Leftrightarrow C_n \sim C_n'$. We call any member of $\mathscr{C}$ the (lower/upper) capacity (if $\mathscr{C}$ is nonempty).

3. THE SPECTRAL ALGORITHM

3.1. *The Linear Case*

For the linear case $d = 1$, Venkatesh and Psaltis (1989a), and Personnaz, Guyon, and Dreyfus (1985) have shown constructions which effectively shape the *spectrum* of the matrix of interconnection weights to ensure that the given set of memories is stable, while obtaining capacities linear in n. The construction entails a selection of weight matrix, $\mathbf{W}$, such that the memories $\mathbf{u}^\alpha$ are eigenvectors of $\mathbf{W}$ with positive eigenvalues. The basic notion used is that if a matrix $\mathbf{U}$ is of full rank the orthogonal projection of a vector $\mathbf{x}$ into the space spanned by the columns of $\mathbf{U}$ is given by $(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{x}$.

Let $\mathscr{B} \geq 0$ be some fixed margin of operation, and consider a fully interconnected network of degree $d = 1$. Fix $m \leq n$, and let $\lambda^{(1)}, \ldots, \lambda^{(m)} > \mathscr{B}$ be fixed (but arbitrary) positive real numbers. Let $\mathbf{u}^1, \ldots, \mathbf{u}^m \in \mathbb{B}^n$ be an m-set of memories whose components are drawn from a sequence of symmetric Bernoulli trials. To each memory $\mathbf{u}^\alpha$ we associate the positive constant $\lambda^{(\alpha)}$. Let

$$\mathbf{U} = [\mathbf{u}^1 \quad \mathbf{u}^2 \quad \cdots \quad \mathbf{u}^m]$$

be the $n \times m$ matrix of memories, and let Λ be the diagonal matrix

$$\Lambda = \begin{bmatrix} \lambda^{(1)} & 0 & \cdots & 0 \\ 0 & \lambda^{(2)} & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & \lambda^{(m)} \end{bmatrix}.$$

The spectral algorithm formally specifies the matrix of interaction weights, $\mathbf{W} = [w_{ij}]$, according to the following rule:

$$\mathbf{W} = \mathbf{U}\Lambda(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T. \tag{4}$$

THEOREM 3.1. *For $d = 1$ and any choice of margin $\mathcal{B} \geq 0$, the spectral algorithm has capacity $C_n = n$.*

In fact, if the prescription (4) yields well-defined weights, then we have

$$\mathbf{W}\mathbf{u}^\alpha = \lambda^{(\alpha)}\mathbf{u}^\alpha.$$

Each memory component is multiplied by a positive scalar, $\lambda^{(\alpha)} > \mathcal{B}$, so that the memories are fixed points under evolution according to the rule (2). As a linear transformation can have at best n eigenvectors with distinct eigenvalues, it follows that n is an upper sequence of capacities for the algorithm. The fact that n is, in fact, the capacity of the algorithm will follow if the prescription (4) is well defined for $m \leq n$ with arbitrarily high probability for large n. This is established by a new result of Kahn, Komlós, and Szemerédi (1990). (This is a refinement of the basic result proved by Komlós in 1967.)

PROPOSITION 3.2. *Almost all $n \times n$ matrices with ± 1 components have full rank; more precisely, if the components of a random $n \times n$ matrix, A_n, are chosen independently and with equal probability $\frac{1}{2}$ from ± 1, then there is a constant $1 < b < 2$ such that the probability that A_n is nonsingular is $1 - O(b^{-n})$.*

The spectral rule amounts (in synchronous operation) to iteratively projecting states orthogonally into the linear space generated by $\mathbf{u}^1, \ldots, \mathbf{u}^m$, and then taking the closest point on the hypercube to this projection. While the algorithm appears to be non-Hebbian and nonlocal, nonetheless, a low complexity, recursive, local construction can be shown for the algorithm using Greville's theorem; the algorithm is, hence, attractive as an associative memory as it combines relatively low complexity with high capacity and efficient error-correction (Venkatesh and

Psaltis, 1989a). This approach can be extended to higher-orders as we now describe.

3.2. *Generalization to Higher-Order*

Let us consider the degree of interaction d to be odd for definiteness. By combining terms we can replace the summation, $\sum_{I \in \mathcal{S}_d} w_{(i,I)} u_I$, for each $i = 1, \ldots, n$ in the evolution rule (2) by an equivalent sum of the form

$$\sum_{k \text{ odd}}^{d} \sum_{1 \le i_1 < \cdots < i_k \le n} w_{i,i_1,\ldots,i_k} u_{i_1} \cdots u_{i_k}. \tag{5}$$

For $\mathbf{u} \in \mathbb{B}^n$ to be a fixed point under evolution according to the rule (2) it, hence, suffices that

$$u_i \sum_{k \text{ odd}}^{d} \sum_{1 \le i_1 < \cdots < i_k \le n} w_{i,i_1,\ldots,i_k} u_{i_1} \cdots u_{i_k} > \mathcal{B}, \qquad i = 1, \ldots, n. \tag{6}$$

Now, for any $\mathbf{u} \in \mathbb{B}^n$ let us define the *kth generation of* $\mathbf{u}$ to be the vector $\mathbf{u}[k] \in \mathbb{B}^{\binom{n}{k}}$ defined by

$$\mathbf{u}[k] = \begin{pmatrix} u_1 u_2 \cdots u_{k-1} u_k \\ u_1 u_2 \cdots u_{k-1} u_{k+1} \\ \vdots \\ u_{n-k+1} u_{n-k+2} \cdots u_{n-1} u_n \end{pmatrix}; \tag{7}$$

in other words, $\mathbf{u}[k]$ is the vector formed by lexicographically ordering the $\binom{n}{k}$ products of components of $\mathbf{u}$ taken k at a time. We now form the vector $\hat{\mathbf{u}}$ from the first $\lceil d/2 \rceil$ odd generations of $\mathbf{u}$:

$$\hat{\mathbf{u}} = \begin{pmatrix} \mathbf{u}[1] \\ \mathbf{u}[3] \\ \vdots \\ \mathbf{u}[d] \end{pmatrix}. \tag{8}$$

Now set

$$N_d = \sum_{k \text{ odd}}^{d} \binom{n}{k}.$$

Clearly, $\hat{\mathbf{u}}$ is a binary vector with N_d components. Let $\hat{\mathbf{W}}$ denote the $n \times N_d$ matrix of coefficients, $w_{i,i_1,\cdots,i_k}$, in Eq. (5) arranged lexicographically; i.e.,

$$\hat{\mathbf{W}} = \begin{bmatrix} w_{11} & \cdots & w_{1n} & w_{1123} & \cdots & w_{1,n-2,n-1,n} & w_{112345} & \cdots & w_{1,n-d+1,n-d+2,\ldots,n-1,n} \\ w_{21} & \cdots & w_{2n} & w_{2123} & \cdots & w_{2,n-2,n-1,n} & w_{212345} & \cdots & w_{2,n-d+1,n-d+2,\ldots,n-1,n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ w_{n1} & \cdots & w_{nn} & w_{n123} & \cdots & w_{n,n-2,n-1,n} & w_{n12345} & \cdots & w_{n,n-d+1,n-d+2,\ldots,n-1,n} \end{bmatrix}.$$

Let $\mathbf{U}$ be the $n \times m$ matrix of memories. Form the extended $N_d \times m$ binary matrix

$$\hat{\mathbf{U}} = [\hat{\mathbf{u}}^1 \quad \hat{\mathbf{u}}^2 \quad \cdots \quad \hat{\mathbf{u}}^m],$$

where $\hat{\mathbf{u}}^\alpha \in \mathbb{B}^{N_d}$ is as defined above. Let

$$\mathbf{\Lambda} = \mathbf{dg}[\lambda^{(1)}, \ldots, \lambda^{(m)}]$$

be an $m \times m$ diagonal matrix with positive diagonal terms, $\lambda^{(\alpha)} > \mathcal{B}$. We formally define the generalized spectral matrix of coefficients, $\hat{\mathbf{W}}$, by

$$\hat{\mathbf{W}} = \mathbf{U}\mathbf{\Lambda}(\hat{\mathbf{U}}^T\hat{\mathbf{U}})^{-1}\hat{\mathbf{U}}^T. \tag{9}$$

Note that this yields stable memories as long as the matrix $\hat{\mathbf{U}}$ is full rank. Specifically, if the initial state is one of the memories, $\mathbf{u}^\alpha$, then we obtain

$$\hat{\mathbf{W}}\hat{\mathbf{u}}^\alpha = \lambda^{(\alpha)}\mathbf{u}^\alpha.$$

It is now easy to verify that Eq. (6) is satisfied for each component of memory $\mathbf{u}^\alpha$, so that $\mathbf{u}^\alpha$ is a fixed point under evolution according to the rule (2). If the degree of interaction, d, is even, the exposition follows as above with the first sum in Eq. (5) being over even k instead of odd k. The maximal allowable rate of growth of m with n follows immediately.

THEOREM 3.3. *An upper capacity of the generalized spectral algorithm of degree d is*

$$\sum_{j=0}^{\lfloor d/2 \rfloor} \binom{n}{d-2j}.$$

In particular, if $d = o(n)$ then an upper capacity is $n^d/d!$.

Anecdotal evidence in implementations indicates that the above estimate of upper capacity actually holds as an estimate of capacity as was the case for $d = 1$. There is some theoretical support for this though no complete proof. The main difficulty is that we cannot directly apply Proposition 3.2 to the matrix $\hat{\mathbf{U}}$ as the distribution induced on vertices of $\mathbb{B}^{N_d}$ as we build up generations according to Eqs. (7) and (8) is not uniform—indeed, we can only access 2^n out of the total of 2^{N_d} vertices. Note, however, that any two distinct vectors, $\mathbf{u}$ and $\mathbf{v}$, in $\mathbb{B}^n$ when expanded to vectors $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ in $\mathbb{B}^{N_d}$ according to Eq. (8) become more and more nearly orthogonal as the number of generations increase. In fact, let D be the Hamming distance between $\mathbf{u}$ and $\mathbf{v}$. Then it is easily verified that the Hamming distance, $\hat{D}$, between $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ is given by[3]

$$\hat{D} = \sum_{j \text{ odd}}^{d} \sum_{k \text{ odd}}^{D} \binom{D}{k}\binom{n-D}{j-k}.$$

(If d is even replace the first sum by a sum over even indices, $j = 0, 2, \ldots, d$.) As d increases the vectors $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ approach orthogonality, and in fact, any pair of vectors $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{B}^n$ result in orthogonal vectors $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ in $\mathbb{B}^{2^{n-1}}$ when all odd (or even) generations are included—i.e., when d is equal to n or $n-1$. To verify this note, for instance, that for any Hamming distance $0 < D < n$ between two vectors in $\mathbb{B}^n$ the corresponding Hamming distance $\hat{D}$ between the corresponding vectors in $\mathbb{B}^{2^{n-1}}$ when all odd generations are included is

$$\hat{D} = \sum_{j \text{ odd}}^{n} \sum_{k \text{ odd}}^{D} \binom{D}{k}\binom{n-D}{j-k}$$

$$= \sum_{k \text{ odd}}^{D} \binom{D}{k} \sum_{j \text{ odd}}^{n} \binom{n-D}{j-k}$$

[3] For simplicity we use the convention $\binom{a}{b} = 0$ if $a < b$ or $b < 0$.

$$= 2^{n-D-1} \sum_{k \text{ odd}}^{D} \binom{D}{k}$$

$$= 2^{n-D-1} 2^{D-1}$$

$$= 2^{n-2}.$$

Hence $\langle \hat{\mathbf{u}}, \hat{\mathbf{v}} \rangle = 0$ for any two vectors $\mathbf{u} \neq -\mathbf{v}$ in $\mathbb{B}^n$ when all odd (or even) generations are included. The preceding analysis does not work when $D = n$; i.e., we start with two opposing vertices of the n-cube. However, even in this case note that the generated vectors $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ become orthogonal if we include all even *and* odd generations. Thus, though the *statistical* dependence across components increases with the number of generations included, we may expect a concurrent building up of *linear* independence as the randomly chosen memories, $\mathbf{u}^\alpha$, result in more and more nearly orthogonal vectors $\hat{\mathbf{u}}^\alpha$. We may, hence, expect the nonsingularity probability estimate of Proposition 3.2 to improve for the generated matrices $\hat{\mathbf{U}}$.[4] In particular, let N_d denote the length of the extended vectors $\hat{\mathbf{u}}$ for any choice of degree d (which may depend on n).

CONJECTURE 3.4. *If the number of memories satisfies $m \leq N_d$ then the $N_d \times m$ extended matrix of memories, $\hat{\mathbf{U}}$, is full rank with probability approaching one as $n \to \infty$.*

This, in turn, would yield that the upper capacity estimate of Theorem 3.3 would actually be the estimate of the capacity of the higher-order spectral algorithm of degree d.

4. MAXIMAL CAPACITY

In this section we derive the maximal storage capacity of a higher-order neural network of degree d. The results are independent of any particular choice of algorithm, and depend only on the network architecture—a higher-order neural network of degree d. The maximal capacity, hence, delineates the upper limit on storage that can possibly be achieved by any particular choice of storage algorithm. We use a fundamental result due to Schläfli (1950) enumerating the number of linearly separable dichotomies of m points in N-space.

Let $\mathbf{V} = \{\mathbf{v}^1, \ldots, \mathbf{v}^m\} \subset \mathbb{R}^N$ be an m-set of points in N-space.

[4] The estimate of Proposition 3.2 may itself be rather weak. As conjectured by Komlós, we may expect the majority of singular ± 1 matrices to be singular for the trivial reason that two rows or two columns coincide. If verified, this would, of course, improve the estimate of the probability of nonsingularity in Proposition 3.2 to $1 - O(n^2 2^{-n})$.

DEFINITION 4.1. A dichotomy $\mathcal{V} = \{\mathbf{V}^+, \mathbf{V}^-\}$ of $\mathbf{V}$ is *homogeneously linearly separable (hls)* if there is a vector $\mathbf{w} \in \mathbb{R}^N$ such that the inner product

$$\langle \mathbf{w}, \mathbf{v} \rangle \begin{matrix} > 0 & \text{if } \mathbf{v} \in \mathbf{V}^+ \\ < 0 & \text{if } \mathbf{v} \in \mathbf{V}^- \end{matrix}. \tag{10}$$

If Eq. (10) holds then $\mathbf{w}$ is said to be a *separating vector* for the dichotomy.

The following version of Schläfli's counting lemma estimates the probability that a randomly chosen dichotomy is homogeneously linearly separable. We give the proof for completeness. The presentation follows that of Wendel (1962) who utilizes the result in this form in a problem in geometric probability. [See also Cover (1965) for a slightly different approach.]

LEMMA 4.2. *Let $\mathbf{V}$ be an arbitrary m-set of points in $\mathbb{R}^N$, and let $\mathcal{V}$ be a dichotomy of $\mathbf{V}$ chosen independently of $\mathbf{V}$, and with equal probability, 2^{-m}, from the set of dichotomies of $\mathbf{V}$. Then the probability, P_N^m, that $\mathcal{V}$ is homogeneously linearly separable is bounded by*

$$P_N^m \leq 2^{-(m-1)} \sum_{j=0}^{N-1} \binom{m-1}{j}. \tag{11}$$

Moreover, a sufficient condition enabling us to replace the inequality above by equality is that the m-set of points $\mathbf{V}$ be chosen from a joint distribution which is such that $\mathbf{V}$ is in general position—i.e., all subsets of size N are linearly independent—with probability one.

Proof. Let D_N^m be the maximum number of dichotomies of an m-set of points in $\mathbb{R}^N$ that are hls. Then

$$P_N^m \leq 2^{-m} D_N^m.$$

In order to demonstrate the validity of Eq. (11) it suffices, hence, to show that

$$D_N^m = 2 \sum_{j=0}^{N-1} \binom{m-1}{j}. \tag{12}$$

Let $\mathbf{V}$ denote an m-set of points for which D_N^m dichotomies are hls. (Such a set exists as $D_N^m \leq 2^m$ is finite.) Let V^α be the hyperplane orthogonal to $\mathbf{v}^\alpha$. Then D_N^m is the number of path-components in $\mathbb{R}^N \setminus \bigcup_{\alpha=1}^m V^\alpha$ as

each path-component is a maximally connected set of vectors all homogeneously separating the same dichotomy of **V**.

Now consider the effect of deleting the hyperplane V^m. The remaining $m - 1$ hyperplanes determine D_N^{m-1} path-components. These are of two types: (i) those path-components (say Q_1 in number) which have a nonnull intersection with the hyperplane V^m, and (ii) those path-components (say Q_2 in number) which do not intersect V^m. Clearly then, $D_N^{m-1} = Q_1 + Q_2$. With V^m restored it cuts each path-component of type (i) in two, and leaves path-components of type (ii) undisturbed. Hence

$$D_N^m = 2Q_1 + Q_2 = D_N^{m-1} + Q_1.$$

Now the intersection of the Q_1 type (i) components with the hyperplane V^m generates Q_1 path-components in $V^m \setminus \bigcup_{\alpha=1}^{m-1}(V^m \cap V^\alpha)$. As the sets $V^m \cap V^\alpha$ are just the hyperplanes in the $(N - 1)$-dimensional space V^m orthogonal to the projection of the vectors $\mathbf{v}^\alpha$ into V^m, it follows that $Q_1 = D_{N-1}^{m-1}$. Hence

$$D_N^m = D_N^{m-1} + D_{N-1}^{m-1}.$$

This recursion with the obvious boundary conditions

$$D_N^1 = D_1^m = 2$$

yields the solution (12) which can be readily verified by induction.

To complete the proof we need to show that we can replace the inequality in Eq. (11) by equality if the m-set of points **V** is in general position with probability one. This follows immediately, however, from the simple observation that the proof above continues to work to estimate the number of hls dichotomies of any m-set of points which has an attribute which is preserved under projections. ■

We require the following technical result due to Chernoff (1952) which gives bounds for very large deviations in the tails of the binomial distribution.

LEMMA 4.3. *Fix $\frac{1}{2} \leq c < 1$ and let H denote the entropy function*

$$H(x) = -x \log_2 x - (1 - x)\log_2(1 - x) \qquad (0 < x < 1).$$

Let p denote the probability that in M trials of a fair coin the number of successes is greater than or equal to cM. Then

$$p = 2^{-M} \sum_{j=\lceil cM \rceil}^{M} \binom{M}{j} \leq 2^{-[1-H(c)]M}.$$

THEOREM 4.4. $\overline{C_n} = 2\binom{n-1}{d}$ *is a maximal upper capacity for zero-diagonal neural networks of degree d.*

Proof. Let $\mathbf{U} = \{\mathbf{u}^1, \ldots, \mathbf{u}^m\}$ be a randomly specified m-set of memories whose components are generated from a sequence of symmetric Bernoulli trials. If each of the memories is to be $\mathcal{B}$-stable we require to find real coefficients, $w_{(i,I)}$, $I \in \hat{\mathcal{J}}_d$, $i \notin I$ such that for each $i \in [n]$, and $\alpha \in [m]$,

$$u_i^\alpha \sum_{I \in \hat{\mathcal{J}}_d : i \notin I} w_{(i,I)} u_I^\alpha > \mathcal{B}. \tag{13}$$

We first argue that without loss of generality we can restrict attention to a margin $\mathcal{B} = 0$. In fact, if there exist a choice of coefficients, $w_{(i,I)}$, such that

$$u_i^\alpha \sum_{I \in \hat{\mathcal{J}}_d : i \notin I} w_{(i,I)} u_I^\alpha > 0, \qquad i = 1, \ldots, n, \quad \alpha = 1, \ldots, m,$$

then, if $T > 0$ is the smallest of the sums above, the simple expedient of scaling all coefficients $w_{(i,I)}$ by a positive scalar greater than $\mathcal{B}/T$ will result in Eq. (13) being automatically satisfied.

Referring to the evolution rule (3) (with margin $\mathcal{B} = 0$) we see that each higher-order neuron in a zero-diagonal network of degree d realizes a separating plane in $\binom{n-1}{d}$-space. For the memories to be fixed points we hence are required for each $i = 1, \ldots, n$ to find $N = \binom{n-1}{d}$ real coefficients $w_{(i,I)}$, $I \in \hat{\mathcal{J}}_d$, $i \notin I$ such that

$$u_i^\alpha = \operatorname{sgn}\left(\sum_{I \in \hat{\mathcal{J}}_d : i \notin I} w_{(i,I)} u_I^\alpha \right), \qquad \alpha = 1, \ldots, m. \tag{14}$$

Now fix i and let $\mathcal{E}_n^i$ be the event that there is no weight vector $\mathbf{w}_i = [w_{(i,I)}]$ in N-space which separates the dichotomy of the extended m-set of memories, $[u_I^\alpha]$, with components varying over the set of indices $I \in \hat{\mathcal{J}}_d$: $i \notin I$, and $\alpha = 1, \ldots, m$, induced by Eq. (14)—i.e., the partition of the memories according to whether u_i^α is -1 or $+1$. Note that the term u_i^α does not appear anywhere in the sum or in the right-hand side of Eq. (14). As the components u_i^α are drawn from symmetric Bernoulli trials it follows that the dichotomy indicated in Eq. (14) is chosen independently of the extended m-set of memories. By Lemma 4.2 we hence have

$$\mathbf{P}\{\mathcal{E}_n^i\} = 1 - P_N^m \geq 1 - 2^{-(m-1)} \sum_{j=0}^{N-1} \binom{m-1}{j}. \tag{15}$$

Let $\mathcal{P}$ be the probability that there exists a zero-diagonal network of degree d in which the fundamental memories are stable. Then

$$\mathcal{P} = 1 - \mathbf{P}\left\{\bigcup_{i=1}^{n} \mathcal{E}_n^i\right\} \leq 1 - \mathbf{P}\{\mathcal{E}_n^i\}. \tag{16}$$

Set $M = m - 1$ for notational convenience. Using Eq. (15) with the upper bound for $\mathcal{P}$ in Eq. (16) we have

$$\mathcal{P} \leq 2^{-M} \sum_{j=0}^{N-1} \binom{M}{j}.$$

Fix $\lambda > 0$ and choose $M = \lceil 2N(1 + \lambda) \rceil$. Then $N = c_1 M$ where $0 < c_1 < \frac{1}{2}$. Using Lemma 4.3 we hence have

$$\mathcal{P} \leq 2^{-M} \sum_{j=0}^{c_1 M} \binom{M}{j} \leq 2^{-[1-H(c_1)]M} \to 0, \qquad (n \to \infty).$$

Hence $2N + 1$ is a maximal upper capacity, and by Proposition 2.6 so is $2N = 2\binom{n-1}{d}$. ■

A maximal lower capacity of N is readily demonstrated if an independence conjecture similar to the one earlier holds. Fix any index i in $[n]$, and consider an extended set of N memories $[u_I^\alpha]_{I \in \mathcal{I}_d; i \notin I}$, $\alpha \in [m]$, where each extended memory is a binary (± 1) vector of length N. Denote this set of (extended) memories by $\tilde{\mathbf{U}}$.

CONJECTURE 4.5. *The set of extended memories $\tilde{\mathbf{U}}$ is linearly independent with probability approaching one as $n \to \infty$.*

For a choice of $m \leq \binom{n-1}{d}$, $P_N^m = 1$ for almost all choices of m memories by Lemma 4.2 if the above holds. This will yield a lower maximal capacity of $N = \binom{n-1}{d}$. We can, however, hope for more: the following application of a result of Füredi (1986) provides a lower bound for the probability that a dichotomy of a randomly chosen m-set from the vertices of an N-cube is hls.

LEMMA 4.6. *Let an m-set of points be chosen independently from the uniform distribution over the vertices of the binary N-cube, $\mathbb{B}^N$. Then, if $m \leq 2N$, the probability that an arbitrary dichotomy of the m-set of points is hls is bounded below by*

$$P_N^m = 2^{-(m-1)} \sum_{j=0}^{N-1} \binom{m-1}{j} - O(b^{-N}), \qquad (N \to \infty),$$

where $b > 1$ is a fixed constant.

Remarks. The exponentially small order term quoted above is a refinement of Füredi's original estimate of $O(N^{-1/2})$ using Proposition 3.2. The result eschews the general position requirement of Lemma 4.2. Specifically, the upper bound for P_N^m in Eq. (11) is sharp if $m \leq 2N$ and the m-set of points is chosen independently from the uniform distribution on vertices of the N-cube.

Füredi's result makes it appear likely that, in fact, $2N = 2\binom{n}{d-1}$ is the maximal capacity of a zero-diagonal higher-order network of degree d. We again have a situation as in the previous section where we would like to apply the result not to the uniform distribution, but to the distribution corresponding to the dth generation of an m-set generated randomly from the uniform distribution on $\mathbb{B}^n$. If the above lemma continues to hold for this situation, then for $m \leq 2N$ we can replace the estimate (15) in the proof of the theorem above by

$$\mathbf{P}\{\mathscr{C}_n^i\} = 1 - 2^{-M} \sum_{j=0}^{N-1} \binom{M}{j} + O(b^{-N}),$$

where, again, we set $M = m - 1$. Using the union bound we have from Eq. (16) that

$$1 - n\mathbf{P}\{\mathscr{C}_n^i\} \leq \mathscr{P}.$$

Fix $0 < \lambda < \frac{1}{2}$ and choose $M = \lfloor 2N(1-\lambda) \rfloor$. Under the above assumption we then have for $d = o(n)$ that

$$\begin{aligned} \mathscr{P} &\geq 1 - n\left[1 - 2^{-M} \sum_{j=0}^{N-1} \binom{M}{j} + O(b^{-N})\right] \\ &= 1 - n\left[2^{-M} \sum_{j=N}^{M} \binom{M}{j} + O(b^{-N})\right] \\ &= 1 - n\left[2^{-M} \sum_{j=c_2 M}^{M} \binom{M}{j} + O(b^{-N})\right], \end{aligned}$$

where $\frac{1}{2} < c_2 < 1$. As $M = \Omega(n^d)$ and $N \sim n^d/d!$ we then have by Chernoff's large deviation bound (Lemma 4.3) that for a choice of constant $c_3 > 0$

$$\mathcal{P} \geq 1 - n2^{-[1-H(c_2)]M} - O(nb^{-c_3 n^d}) \to 1, \qquad (n \to \infty).$$

So $2N + 1$ is a lower sequence of maximal capacities, and hence, so is $2N$ by Proposition 2.6 if Füredi's result holds in this case.

For the case $d = 1$ it is clear that Füredi's lemma holds *in toto* so that the above analysis works with $N = n - 1$. For the case of linear interactions, hence, we have shown the following

THEOREM 4.7. *The sequence $2n$ is the maximal capacity for zero-diagonal neural networks with linear interactions, $d = 1$.*

Remark. It is known that $2n$ is the capacity of a single linear threshold element [cf., for instance, Cover, 1965; Venkatesh and Psaltis, 1991]. The above result asserts that there is no decrease in capacity for the zero-diagonal network of n neurons even though we now have a situation where n neurons operate on the *same* set of memories.

5. CONCLUDING OBSERVATIONS

1. For the case $d = 1$ Abu-Mostafa and St. Jacques (1985) demonstrate that with the requirement that *all* choices of m vectors be stored as fixed points for some choice of zero-diagonal network, m can be no larger than n. However, small pathological sets of vectors which cannot be stored can be found (Montgomery and Vijayakumar, 1986), and such pathologies make it difficult to achieve nontrivial deterministic capacities. The probabilistic setup adopted here essentially relaxes the requirement that *all* choices of m vectors be storable to the requirement that *almost all* choices of m memories be storable; pathological scenarios that cannot be stored form a set whose size is small compared to $\binom{2^n}{m}$, and are effectively ignored in this definition.

2. The maximal capacities for nonzero diagonal networks are of the same order as those for the zero-diagonal networks. Note, however, that we are required to put restrictions on the allowable choices of interactions. Specifically, consider the case $d = 1$. With a choice of identity matrix of interactions, $w_{ij} = \delta_{ij}$, it is clear that *all* states in $\mathbb{B}^n$ are stable with the same margin of stability. There is clearly no associative storage possible in this situation. To avoid situations of this type we have to put constraints on the allowable interactions so that the number of extraneous stable states do not become too large: specifically, the diagonal terms

should not dominate the nondiagonal terms. Similar examples hold for the higher-order cases.

3. The capacity estimates continue to hold if we are required to store random associations of the form $\mathbf{u}^\alpha \mapsto \mathbf{v}^\alpha$. We then call the vectors $\mathbf{v}^\alpha$ the *associated memories*. The spectral algorithm generalizes in a straightforward manner with the interaction matrix of coefficients of Eq. (9) modified to

$$\hat{\mathbf{W}} = \mathbf{V}\Lambda(\hat{\mathbf{U}}^T\hat{\mathbf{U}})^{-1}\hat{\mathbf{U}}^T$$

with $\mathbf{V}$ being the $n \times m$ matrix of associated memories.

4. The main unresolved issue in this work is the conjecture introduced in this paper that the linear independence property is preserved (strengthened!) when we consider higher generations of vectors chosen uniformly from $\mathbb{B}^n$. This is independent of the Komlós conjecture.

ACKNOWLEDGMENTS

We are grateful to the anonymous referee for his suggestions toward the improvement of the paper, and, in particular, for his comments on the linear independence issue. This work was supported in part by NSF Grants EET-8709198 and DMS-8800322, ONR Contract 411P006-01, and by Air Force Grant AFOSR 89-0523.

REFERENCES

ABU-MOSTAFA, Y. S., AND ST. JACQUES, J. (1985), Information capacity of the Hopfield model, *IEEE Trans. Inform. Theory* **IT-31,** 461–464.

BALDI, P., AND VENKATESH, S. S. (1987), Number of stable points for spin glasses and neural networks of higher orders, *Phys. Rev. Lett.* **58,** 913–916.

BALDI, P., AND VENKATESH, S. S. (1988), On properties of networks of neuron-like elements, *in* "Neural Information Processing Systems" (D. Z. Anderson, Ed.), Amer. Inst. Phys., New York.

BALDI, P. (1988), Neural networks, orientations of the hypercube, and algebraic threshold functions, *IEEE Trans. Inform. Theory* **IT-34,** 523–530.

CHERNOFF, H. (1952), A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations, *Ann. Math. Statist.* **23,** 493–507.

COVER, T. M. (1965), Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition, *IEEE Trans. Elec. Comput.* **EC-14,** 326–334.

FÜREDI, Z. (1986), Random polytopes in the *d*-dimensional cube, *Discrete Comput. Geom.* **1,** 315–319.

GOLES, E., AND VICHNIAC, G. Y. (1986), Lyapunov functions for parallel neural networks, *in* "Neural Networks for Computing" (J. Denker, Ed.), Amer. Inst. Phys., New York.

HOPFIELD, J. J. (1982), Neural networks and physical systems with emergent collective computational abilities, *Proc. Natl. Acad. Sci. USA* **79,** 2554–2558.

KAHN, J., KOMLÓS, J., AND SZEMERÉDI, E. (1990), Singularity probabilities for random ± 1 matrices, preprint.

LEE, Y. C., DOOLEN, G., CHEN, H. H., SUN, G. Z., MAXWELL, T., LEE, H. Y., AND GILES, C. L. (1986), Machine learning using a higher-order correlation network, *Physica* **22D,** 276–306.

MAXWELL, T., GILES, C. L., LEE, Y. C., AND CHEN, H. H. (1986), Non-linear dynamics of artificial neural systems, *in* "Neural Networks for Computing" (J. Denker, Ed.), Amer. Inst. Phys., New York.

MCCULLOCH, W. W., AND PITTS, W. (1943), A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.* **5,** 115–133.

MONTGOMERY, B. L., AND VIJAYAKUMAR, B. V. K. (1986), Evaluation of the use of the Hopfield neural network model as a nearest neighbour algorithm, *Appl. Opt.* **25,** 3759–3766.

PERSONNAZ, L., GUYON, I., AND DREYFUS, G. (1985), Information storage and retrieval in spin-glass like neural networks, *J. Physique Lett.* **46,** L359–L365.

PSALTIS, D., AND PARK, C. H. (1986), Nonlinear discriminant functions and associative memories, *in* "Neural Networks for Computing" (J. Denker, Ed.), Amer. Inst. Phys., New York.

SCHLÄFLI, L. (1950), "Gesammelte Mathematische Abhandlungen I," pp. 209–212, Verlag Birkhäuser, Basel, Switzerland.

VAPNIK, V. N. (1982), Estimation of dependences based on empirical data, *in* "Springer Series in Statistics," Springer, New York/Berlin.

VENKATESH, S. S. (1986), "Linear Maps with Point Rules: Applications to Pattern Classification and Associative Memory," PhD thesis, California Institute of Technology.

VENKATESH, S. S., AND PSALTIS, D. (1989), Linear and logarithmic capacities in associative neural networks, *IEEE Trans. Inform. Theory* **IT-35,** 558–568.

VENKATESH, S. S., AND BALDI, P. (1989), Random interactions in higher-order neural networks, submitted for publication.

VENKATESH, S. S., AND BALDI, P. (1991), Programmed interactions in higher-order neural networks: The outer-product algorithm, *J. Compl.*, to appear.

VENKATESH, S. S., AND PSALTIS, D. (1991), On reliable computation with formal neurons, *IEEE Trans. Pattern Anal. Mach. Intell.*, to appear Aug. 1991.

WENDEL, J. G. (1962), A problem in geometric probability, *Math. Scand.* **11,** 109–111.