R.J. Marks II, L.E. Atlas and K.F. Cheung, "Architectures for a continuous level neural network based on alternating orthogonal projections", Proceedings of O-E/LASE '88 Conference on Neural Network Models for Optical Computing, Los Angeles, January 1988, SPIE volume 882, pp 90-92.

ARCHITECTURES FOR A CONTINUOUS LEVEL NEURAL NETWORK BASED ON ALTERNATING ORTHOGONAL PROJECTIONS

Robert J. Marks II, Les E. Atlas, Seho Oh and Kwan F. Cheung

Interactive Systems Design Laboratory, FT-10
University of Washington, Seattle, WA 98195

ABSTRACT

Optical processor architectures for various forms of the alternating projection neural network (APNN) are considered. Required iteration is performed by passive optical feedback using only free space and guided propagation. No electronics or slow optics (e.g. phase conjugators) are used. The processor can be taught a new training vector by viewing it only once.

INTRODUCTION

Optical neural network architectures have been proposed by a number of researchers [1-5]. Using the alternating projection neural network (APNN) algorithm developed elsewhere [6-9], we propose two architectures for a corresponding optical implementation. Required iteration is accomplished using only guided or free space propagation. No electronics or slow optics such as phase conjugators are used in the feedback path. Learning can be performed by viewing each training vector only once. The network has been shown to scale well [6-9]. The number of training vectors that can be stored is on the order of the total number of neurons minus the number of floating (or output) neurons. The APNN's storage capacity can be increased by the use of additional neurons in the hidden layer.

PRELIMINARIES

In this section, we present a terse overview of the APNN. More detailed explanations are available elsewhere [6-9].

Consider a set of N continuous level library vectors of length $L > N$: $\{f_n \mid 0<n<N\}$. We form the library matrix $\mathbf{F}=[f_1 \mid f_2 \mid \ldots \mid f_N]$ and the interconnect matrix

$$\mathbf{T} = \mathbf{F}\,(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T$$

The L neurons are divided into two sets: one in which the neural states are known and the remainder in which the states are unknown. Let $S_k(m)$ denote the state of the kth neuron at time m. If the kth neuron falls into the known category, then its state is *clamped* to the known value. Otherwise the state remains *floating* and is equal to the sum of the neural inputs. Assume without loss of generality that the states of neurons 1 through $P<L$ are known and the remaining $Q=L-P$ are not. Let f denote a library vector of which we know the first P elements. Define the corresponding node operator, N, on an arbitrary vector i by

$$Ni = N\,[i_1 \;\; i_2 \;\; i_P \mid i_{P+1} \;\; i_L]^T$$
$$= [f_1 \;\; f_2 \;\; f_P \mid i_{P+1} \;\; i_L]^T$$

where f_k is the kth element of f. Then, in synchronous form, the network performs the operation

$$S(m+1) = N\,\mathbf{T}\,S(m) \qquad (1)$$

where $S(m)$ is the vector of neural states at time m. Convergence to f is assured if the first P rows of $\mathbf{F}$ form a matrix of full column rank. Subsumed in this is the criterion that $P>N$.

LAYERED IMPLEMENTATION

By *layered*, we mean that the same neurons are always used to stimulate the network and the same set is always floating. In addition, there is a *hidden* layer of neurons whose principal purpose is to increase the storage capacity of the network. Use of a hidden layer also increases the convergence rate of the network and decreases its sensitivity to the inexactness of analog multiplication [9].

In order for the APNN to converge to the proper solution, the number of clamped states must equal or exceed the number of stored library vectors. The number of clamped neural states, however, can be artificially increased by using new *hidden* neurons the states of which are a function of the known portion of the input library vector. The hidden states can, in general, be any nonlinear combination of the clamped states. A technique commonly used with neural networks is to run a linear combination of the clamped states through a sigmoidal nonlinearity to determine the hidden states. Alternately, products of clamped states could be used [10]. Once established, the hidden states are treated as clamped states in the previously discussed analysis. Although the choice of the nonlinearity does not affect the response of the network to training data, it does affect the manner in which the network generalizes.

A basic architecture for optical implementation of the layered APNN is shown in Figure 1. The point source array elements corresponding to the clamped and hidden layers provide the input to a Stanford matrix-vector multiplier (astigmatic focusing optics are not shown) [11]. The point source array for the floating layer is used only when training the network. The same is true for the detector array at the output. Indeed, the only neurons of interest are the floating ones. The states corresponding to the floating neurons are introduced at the right to a fiber bundle. These intensities are fed back to the input as shown and the process corresponding to (1) is repeated iteratively until convergence. Alternately, mirrors can be used to provide the feedback [12].

The astute reader will have noticed two implementation problems associated with the architecture in Figure 1. First, there is no provision to detect the output. Second, there is no apparent provision for compensating for absorbtive, coupling and other losses in the feedback path. Each of these problems has a straightforward solution. To detect the output, we simply place a highly transmitting pellicle in the feedback path and focus the reflected portion onto a detector array (not shown). This contributes more to

the problem concerning losses in the feedback path.

The magnitude of the elements of **T** are generally quite small when $N<<L$. For example, if the library matrix consists of only plus and minus ones and the library vectors are orthogonal, then the maximum value of the magnitude of the elements in **T** is N/L. (We would expect such orthogonal library vectors in the statistical sense if all elements of **F** were chosen by a 50-50 coin flip). When each element of **T** is small, feedback losses can be compensated by scaling the intensity transmittance up to its maximum passive value of unity. Note then that the storage capacity of the network is then in part a function of the ability to minimize feedback losses.

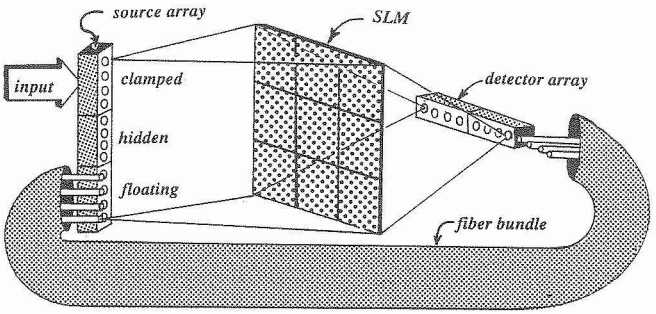

Figure 1: An architecture for performing a layered APNN neural network. In practice, the architecture requires augmentation as in [11] to allow for the required bipolar operations. The states of the hidden layers are nonlinear functions of the clamped layers and are generated electronically.

HOMOGENEOUS IMPLEMENTATION

When every neuron in the network can either be clamped or floating, the APNN is said to be *homogeneous*. In this form, the network stimulus can be provided by different neurons from application to application.

An architecture for a homogeneous APNN is shown in Figure 2. Clamped neural states are provided by the point source array. The five darkened dots on the array correspond to the five clamped neural states in this example. These provide the input vector intensities for the portion of the Stanford matrix-vector multiplier corresponding to the upper spatial light modulator (SLM). The light is collected at the output and, as before, is fed back through a fiber optics bundle to the input. The light from the fiber output is fed through the lower portion of the Stanford matrix-vector multiplier whose SLM transmittance compensates for feedback loss. The processor output is detected as before.

Note that there should be no input from fibers corresponding to clamped neurons. This can be accomplished with either a optic-optic or electro-optic toggle that turns off the fibers corresponding to the locations of the clamped neurons. Such switches can operate in the gigahertz range with small attenuation [13].

TRAINING

The interconnect matrix equation in (1) for most cases is computationally unacceptable. In the spirit of learning, the interconnect matrix can be constructed one vector at a time using a Gram-Schmidt procedure[7-8]. If, for example, we wish to include a new library vector f in an established APNN with an interconnect matrix **T**, the revised interconnect matrix is

$$\mathbf{T}_+ = \mathbf{T} + e\, e^T / (e^T e) \qquad (2)$$

where $e = (\mathbf{I} - \mathbf{T})f$. A training vector can be forgotten by subtracting rather than adding.

Consider, then, training the homogeneous APNN in Figure 2. A new training vector f is input on the source array. Since the fibers are turned off, the vector $\mathbf{T}f$ will be read by the output detector (not shown). The output is subtracted electronically from the input to give the vector e. The SLM is then updated in accordance to (2).

If the new library vector f is a linear combination of the previous library vectors, then e will contain all zeros. Due to the computational inexactness of analog computations in such a case, e will be close to but not exactly equal to the zero vector. This motivates us to compare the energy e^Te to a small threshold prior to decide whether or not (2) should be applied.

The layered APNN in Figure 1 can be similarly trained. As with the case above, the input to the system from the fibers is suppressed. The vector f is input on the source array. The output is read by the detector array shown and the detector array used to read the output in the recall mode (not shown). Updating the SLM transmittance is done as before. The energy of the error corresponding to the output neurons only, however, should be used to determine whether or not to use (2).

Note that for both the layered and homogeneous case, a significant portion of the SLM is used only in the learning process.

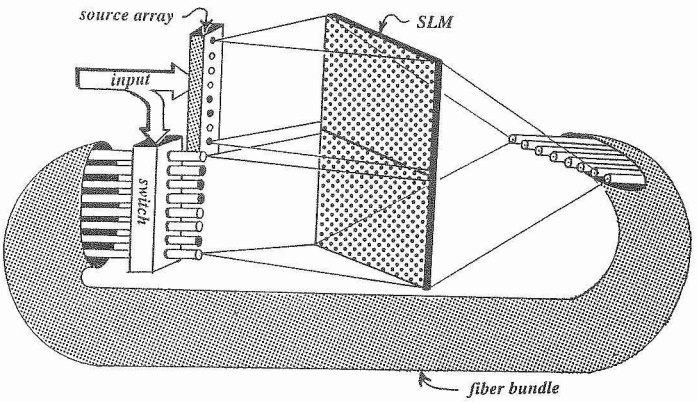

Figure 2: An architecture for performing a homogeneous APNN.

REFERENCES

1. N. Farhat, D. Psaltis, A. Prata and E. Paek, "Optical implementation of the Hopfield model" **Applied Optics**, vol.24, p.1469 (1985).

2. D. Psaltis and N. Farhat, "Optical information processing based on an associative memory model of neural nets with thresholding and feedback", **Optics Letters**, vol.10, p.98 (1985).

3. B. Macukow and H.H. Arsenault "Optical associative memory model based on neural networks having variable interconnection weights", **Applied Optics**, vol.26, p.924 (1987).

4. N. Farhat, "Architectures for optoelectronic analogs of self organizing neural networks", **Optics Letters**, vol. 12, p.448 (1987).

5. Special issue on optical artificial intelligence, **Applied Optics**, vol. 26 (1987)

6. R.J. Marks II, "A Class of Continuous Level Associative Memory Neural Nets," **Appl. Opt.**, vol.26, no.10, pp.2005-2010, 1987.

7. K.F. Cheung, S. Oh, R.J. Marks II and L.E. Atlas, "Neural Net associative Memories Based on Convex Set Projections,: **Proc. IEEE First International Conference on Neural Networks**, San Diego, 1987.

8. R.J. Marks II, L.E. Atlas and K.F. Cheung "A class of continuous level neural nets", **Proceedings of the Fourteenth Congress of the International Commission for Optics**, pp.29-30, Quebec City, Quebec Canada, August 24-28, 1987.

9. R.J. Marks II, L.E. Atlas, S. Oh and J.A. Ritcey "The performance of convex set projection based neural networks", **Neural Information Processing Systems - Natural and Synthetic**, (American Institute of Physics Press, 1988).

10. T. Maxwell, C.L. Giles and Y.C. Lee, "Transformation invariance using higher order correlations in neural net architectures", **Proc. 1986 IEEE International Conference on Systems, Man , and Cybernetics**, (Atlanta GA, 14-17 October 1986) p.627.

11. J.W. Goodman, A.R. Dias, and L.M. Woody "Fully parallel high speed incoherent optical method for performing discrete Fourier transforms", **Optics Letters**, vol.2, p.1 (1978).

12 R.J. Marks II "Coherent optical extrapolation of two-dimensional signals: processor theory", **Applied Optics**, vol. 19, pp.1670-1672 (1980).

13. see, for example, H. Haga, "$LiNbO_3$ travelling-wave light modulator/switch with an etched groove", **IEEE J. Quan. Electronics**, vol.QE-22, p.902 (1986).

ACKNOWLEDGEMENT

This work was supported by the Washington Technology Center and the Boeing High Technology Center in Bellevue, WA. Les E. Atlas was also supported by an NSF Presidential Young Investigator Award.