

# Systolic Array Implementation of Block Based Hopfield Neural Network for Pattern Association

Ming-Jung Seow, Hau Ngo, Vijayan Asari  
 Department of Electrical and Computer Engineering  
 Old Dominion University, Norfolk, VA, 23529, USA  
 E-mail: mseow@odu.edu, hngox001@odu.edu, vasari@odu.edu

## Abstract

*This paper suggests the systolic array implementation of block based Hopfield neural network architecture using completely digital circuits. The design is based on rewriting the energy equation of Hopfield neural network to a systolic (or modular) form. The performance of the proposed architecture is evaluated by applying various binary inputs and it is observed that the network provides massive parallelism and can be extended by cascading identical chips.*

## 1. Introduction

Computations in Artificial Neural Network models are based on the same organizational principles as the brain. Just as in the brain, Artificial Neural Network models employ many simple computational elements that work concurrently to achieve brain-like tasks. Although Artificial Neural Network have shown great promise, their full potential has yet to be realized as most implementations have been on sequential machine that are unable to exploit the inherent parallelism in these networks. Thus, dedicated chips implementing the neural network in the VLSI form is required to realize the full capability of neural network [1].

In this paper, we propose a systolic array implementation of block based Hopfield neural network architecture [2] using fully digital circuit. The proposed architecture requires pre-computed synaptic weights to perform computation for recognition of the input patterns.

## 2. Systolic array implementation of a block based Hopfield neural network

The block based Hopfield neural network proposed by Seow and Asari [2] is a recurrent neural network with each module of neurons working independently. The

weights [3] could be modified by incorporating the D factor [2] as:

$$w_{ijkl} = \sum_{s=1}^P x_{ij}^s x_{kl}^s D_{ijkl} \quad (1)$$

for  $1 \leq i, k \leq N$  and  $1 \leq j, l \leq M$

where  $D_{ijkl}$  is the distance from the  $(k, l)^{th}$  neuron to the  $(i, j)^{th}$  neuron. The D factor is a controlling parameter in which the relationship between each group of neurons can be distinguished. By specifying the point of reference of each neuron with respect to every other neuron, the D factor can be used for modeling the architecture. That is, the D factor could be used to redistribute the energy function to create a new modular architecture. The net output of the network is computed as:

$$Net_{ij} = \sum_{k=1}^N \sum_{l=1}^M w_{ijkl} x_{kl} \quad (2)$$

for  $1 \leq i \leq N$  and  $1 \leq j \leq M$

and the output is thresholded as:

$$o_{ij} = f(Net_{ij}) = \begin{cases} +1 & \text{if } Net_{ij} \geq 0 \\ -1 & \text{if } Net_{ij} < 0 \end{cases} \quad (3)$$

for  $1 \leq i \leq N$  and  $1 \leq j \leq M$

## 2.1. Network architecture based on distance factor

Let the input image of size  $N \times M$  be divided into sub-blocks of size  $n \times m$  and let each sub-image applied to one of the NM/nm sub-networks with each module working independently. It can be noticed that an  $n \times m$  sub-image area and its neighbors in a larger image of size  $N \times M$  have very similar intensity values. Let the value of the distance factor D for a cellular structure with cell size of  $n \times m$  be expressed as [2]:

$$D_{ijkl} = \begin{cases} 1 & \text{if } \eta = 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

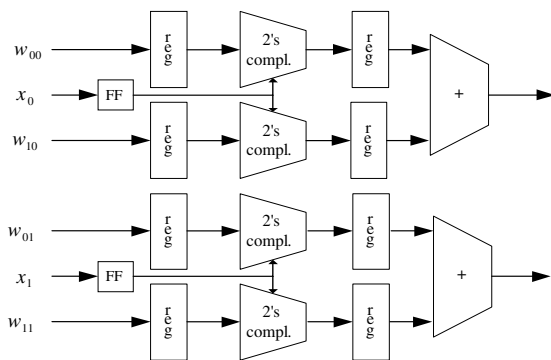
where  $\eta$  is defined by [2]:

$$\eta = \max \left\{ \left| \left\lfloor \frac{i-1}{n} \right\rfloor - \left\lfloor \frac{k-1}{n} \right\rfloor \right|, \left| \left\lfloor \frac{j-1}{m} \right\rfloor - \left\lfloor \frac{l-1}{m} \right\rfloor \right| \right\} \quad (5)$$

It can be observed that the influence of the neuron in the second cell-neighborhood onwards on a particular neuron under consideration will be reduced significantly. One of the major advantages of the distance (or D) based training algorithm is that all the weights located outside the boundary D are not necessary to be trained. This considerably reduces the number of weights to be trained and hence the training time.

### 3. Architecture of Hopfield neural network

A detailed architecture for each module with two neurons is shown in Figure 1 where  $x_j$  represents the  $j^{\text{th}}$  bit of the pattern and  $w_{ij}$  is a weight obtained during training process. Inputs weights are latched into registers (*reg*) before feeding them to the 2's complement units (*2's compl.*), which has the functionalities: (1) pass the data from input to output when controlling signal  $x_j=0$  or (2) negate input and output the result when controlling signal  $x_j=1$ , for synchronizing the data. Registers are introduced between two-complement units and adders to reduce output delay at each adder. For the purpose of demonstrating our technique, the architecture for an 8-bit neural network is designed and implemented. The network consists of eight neurons, and each neuron is responsible for a single bit in the pattern.



**Figure 1: Each module of neuron architecture**

The complexities of modular structures related to non-modular structures are well studied. For instance, three possible complexity discriminations of modular structures with respect to non-modular structures can be the neuron counting approach in which the complexity of the network is proportional to the number of neurons in the network, the weight counting approach in which the neurons of the network are all labeled a certain weight and networks are compared based on their total weights, and finally the dimensional approach in which all neurons of the network take up some space and networks

are compared based on the minimum possible volume of the space containing the network. It can be observed that the systolic array implementation of modular Hopfield neural network shows less complex overall compare to the conventional way of implementing Hopfield neural network

### 4. Simulation results and performance evaluation

For the performance evaluation of the designed Hopfield neural network, the 8-bit architecture for the neural network is implemented and simulated with Altera's Quartus II version 1.1 design tool. The entire architecture is fitted into a FPGA from Altera's APEX family. Specifically, the EP20K60EFC324-1X FPGA is selected for compilation and simulation purposes. Based on simulation results, the maximum clock frequency that can be applied to the network is 83 MHz. The total number of logic cells that the entire architecture resides on is 1941, and the total number of flip-flops used for this design is 1696. The latency of each Neuron is 2 clock cycles for each input; the results of these neurons are then summed and produce the  $Net_j$ .

### 5. Conclusion

A new systolic array implementation of Hopfield neural network using completely digital circuit is described. The design is based on rewriting the energy equation of Hopfield neural network to facilitate the change to systolic form. In addition, it can be shown that the architecture provides massive parallelism and can be extended by cascading identical chips.

### 6. Reference

- [1] K. V. Asari and C. Eswaran, "Systolic Array Implementation of Artificial Neural Networks", *Journal of Microprocessors and Microsystems*, vol. 18, no. 8, pp. 481-488, Oct. 1994.
- [2] Ming-Jung Seow and Vijayan K. Asari, "High storage capacity architecture for pattern recognition using an array of Hopfield Neural Networks", *IEEE Computer Society Proceedings of 30th International Workshop on Applied Imagery and Pattern Recognition, AIPR - 2001*, Washington DC, USA, pp. 169 - 174, Oct. 10 - 12, 2001.
- [3] J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities", *Proc. Natl. Acad. Sci.* 79, 1982