Artificial Neural Networks Using Complex Numbers and Phase Encoded Weights

Howard E. Michel and Abdul Ahad S. Awwal

hmichel@umassd.edu
aawwal@cs.wright.edu

Electrical and Computer
Engineering Department
UMass Dartmouth
285 Old Westport Road
N. Dartmouth MA 02747

Department of Computer
Science and Engineering
Wright State University
Col. Glen Highway
Dayton OH 45435

Abstract

The model of a simple perceptron using phase-encoded inputs and complex-valued weights is proposed. The aggregation function, activation function, and learning rule for the proposed neuron are derived and applied to Boolean logic functions and simple computer vision tasks. The complex-valued neuron (CVN) is shown to be superior to traditional perceptrons. An improvement of 135% over the theoretical maximum of 104 linearly separable problems (of three variables) solvable by conventional perceptrons is achieved without additional logic, neuron stages, or higher order terms such as those required in polynomial logic gates. Use of the CVN in character recognition and image segmentation is demonstrated. Implementation details are discussed and shown to be very attractive for optical implementation since optical computations are naturally complex.

Introduction

The processing power of an artificial neuron is dependent on the information representation used in the neuron. Traditionally, artificial neural networks (ANNs) have relied on real-valued weights. The interconnection weights—which represent the learned behavior of the ANN—are derived from the recognition that at a simplified level, a biological neuron's firing rate represents the information in the network. However, some of the limitations of existing ANNs may be traced to the limitations in the representation of information. Figure 1 illustrates an artificial neuron model.

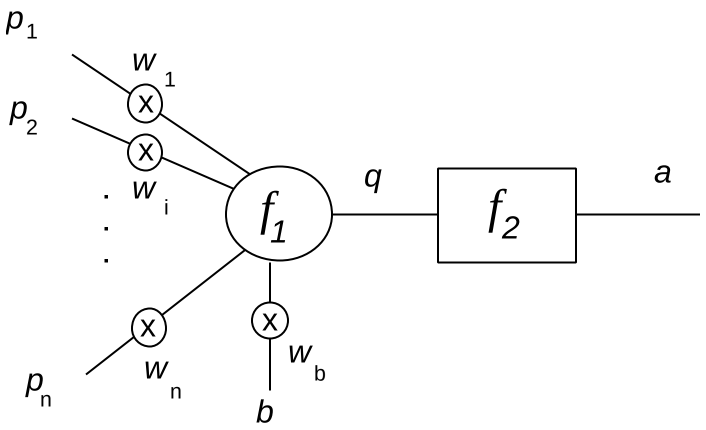


Figure 1 Representation of an artificial neuron

This research builds on the similarity between a spiking neuron, where input data is represented by time-encoded impulse trains and traditional electronic signal analysis where information is represented by magnitude and phase components. It is thought then, that representing real world digitized scalar data as phase information in ANNs might improve the performance of ANNs. This thought is extended to using the angle portion of a complex number to represent the phase of a periodic signal with amplitude represented by the magnitude of the complex number.

The work here extends complex numbers for general ANN. It is shown that representing real world digitized scalar data as phase and operating on this data in the complex-domain improves the performance of ANNs. The representation of the new neuron is shown to be at least as computationally powerful as, and in many cases more powerful than existing ANNs.

Development

Mathematical representation of the proposed neuron

The proposed complex-valued artificial neuron is similar in composition to a traditional artificial neuron except all weights, w_i, will be represented by complex numbers. Externally, input data and output data will be real.

The ***input mapping*** defines how the real-world data will be represented in the ANN calculations. To express the input mapping for the complex valued artificial neuron, assume that the set of input variables P is composed of n-tuples $\mathbf{p}_i$, where each of the components p_i is expressed as (1). Equation (2) is the version of (1) for $\lambda_p = 1$ that maps any continuous value in the range

0 → max_value into the complex inputs required for the CVN. Inputs are thus coded as periodic pulse trains with unity magnitude and different phases. This is a mapping from $\mathsf{R}^n \rightarrow \mathsf{C}^n$.

$$p_i = l_{p_i} e^{iy_i} \qquad (1)$$

$$p_i = \exp\left(i \frac{value}{\mathrm{max\_value}} \frac{p}{2} \right) \qquad (2)$$

The complex-valued ***aggregation function*** is designed after the form of a traditional neuron's aggregation function as shown in (3). Here, $\mathbf{p} \in \mathrm{C}^{\mathrm{n}}$ is a column vector of the input components p_i, and $\mathbf{w} \in \mathrm{C}^{\mathrm{n}}$ is a row vector of weights terms w_i. The aggregation function is thus a mapping $\mathrm{C}^{\mathrm{n}} \rightarrow \mathrm{C}$.

The aggregation function feeds directly into the ***activation function***. The complex-valued neuron will use a perceptron-like activation function, that is, a hard limiting function. Because the magnitude of complex number is easy to compute, and easy to measure optically and electronically,

$$q = \mathbf{wp} \qquad (3)$$

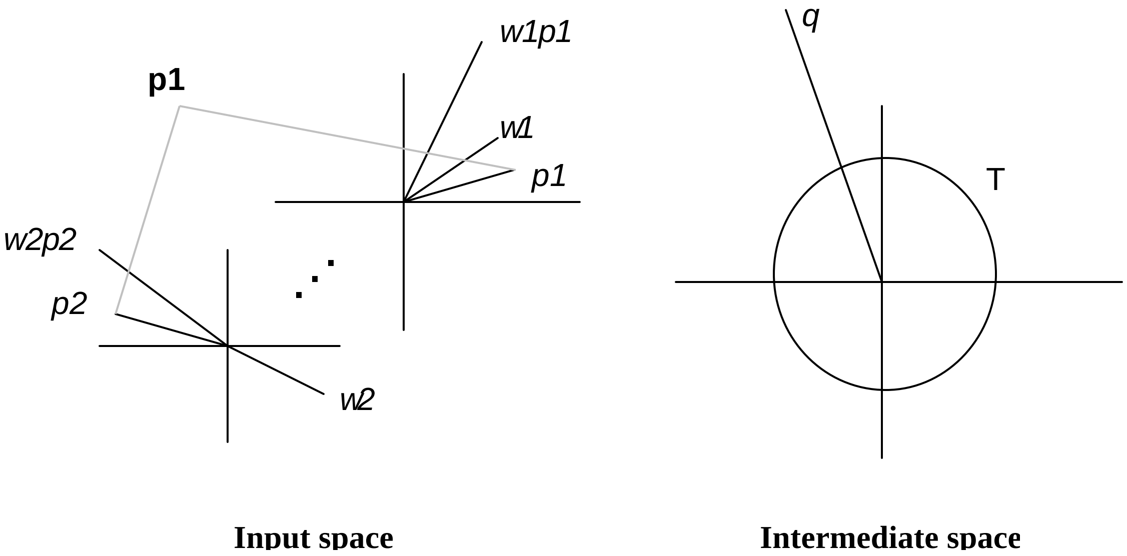
$$a = \begin{cases} 0 \text{ if } |q| < T \\ 1 \text{ if } |q| \geq T \end{cases} \qquad (4)$$

and because it captures the effects of angle differences and individual component magnitudes, it was chosen for the activation function. The activation function is shown in (4),where *a* and *T* are real numbers, and *q* is complex.. This is equivalent to a circular threshold. The activation function mapping is thus of the form C → R.

In a traditional neuron, an ***output mapping*** from an internal representation to the physical representation is required. This is a mapping of the form R → R that typically is concerned with scaling and/or numerical accuracy. Because the complex-valued artificial neuron's activation function is of the form C → R, the output mapping in the complex-valued neuron is of the form R → R and is identical to the traditional neuron's output mapping.

Unlike traditional neurons, this aggregation function is not linear with respect to the real-world data. The resultant output is dependent on the relationships among the various weights and inputs, as well as their individual values.

Figure 2 shows these relationships graphically.



Input space

Intermediate space

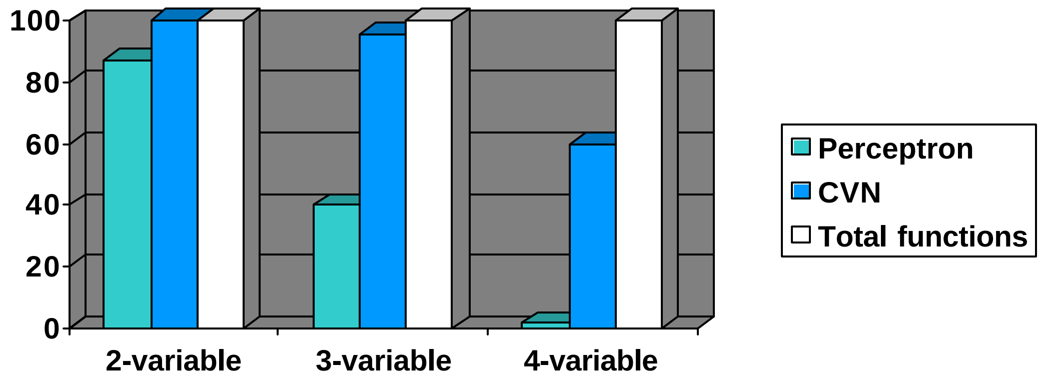
Figure 2 Complex valued neuron data transformation

Results

Computer Simulation of Boolean Logic Problems

The CVN has been used to solve the 2, 3, and 4-input Boolean problems. The CVN is capable of learning all 16 possible functions of two Boolean variables, x_1 and x_2. Traditional perceptrons are capable of learning only 14 of those functions. It is superior in 3 and 4 input problems.

Percent of total functions solveable



Character recognition using complex valued neurons

Character recognition is a common task in computer vision. Often these characters are similar and/or distorted. For example consider the task of differentiating the letter “C” from the letter “T”, as represented by square pixels in a three by three grid. Figure 3 is a depiction of the characters. There is a Hamming distance of two between these two codes because of the difference in the upper left and lower right.

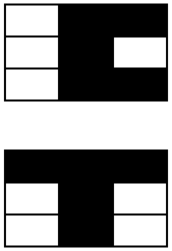


Figure 3 Three by three representation of “C” and “T”

Now consider the identification task if the letters are distorted by only one pixel each. There are nine possible variation of each correct letter. Some of the possible representations of the letter “C” are shown in figure 4, and some possible representations of the letter “T” are shown in figure 5.

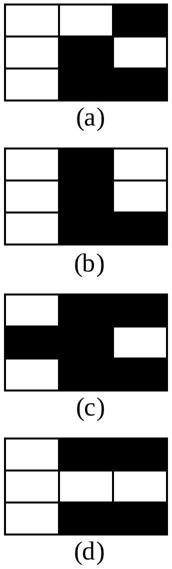

(a)
(b)
(c)
(d)

Figure 4 Variations on the letter “C”

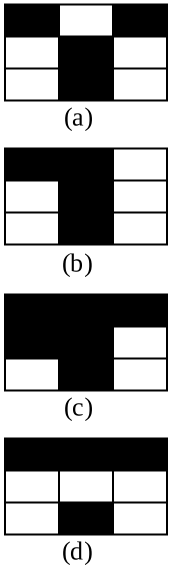

(a)
(b)
(c)
(d)

Figure 5 Variations on the letter “T”

A single 9-input complex valued neuron was trained to recognize the correct representation of the letters “C” and “T.” When trained, the complex valued neuron correctly identified the distorted “C” and “T” characters that were three pixels from the opposite character. The complex valued neuron split evenly on assigning the characters that were only one pixel different. This is an appropriate solution since those characters could have just as likely been originally from either “C” or ”T.”

Region segmentation

Another common task in computer vision is segmenting a digital image into several regions. Often this is done by traditional computer vision methods such as smoothing and thresholding. Pixels above that threshold are assigned into one region and pixels below that threshold are classified into a second region. Methods of selecting an appropriate threshold vary, but often involve calculating a histogram of brightness levels and selecting a value that will divide the histogram into two "natural" areas.

Figure 6 is a picture of a skin-cancer growth depicted in 255 gray levels. The size of the picture is 256 by 256 pixels. It requires a moderate amount of processing—smoothing, calculating a histogram, selecting a threshold, and applying the threshold pixel by pixel—to segment it into two regions.

Selected three-by-three pixel squares were input to the neuron to allow the neuron to simultaneously learn the cancer gray levels as well as the non-cancerous gray levels, and to smooth the pixel level noise within the window. Each neuron is presented with a sequence of 9-tuple input patterns representing both the cancer cell samples and the non-cancer cells.

After training, all 3-pixel by 3-pixel windows of the figure were tested with the trained complex valued neuron operating in a recall phase. The detected cancer cells were recoded as black and negative detection was recoded as white. Figure 7 shows the result.

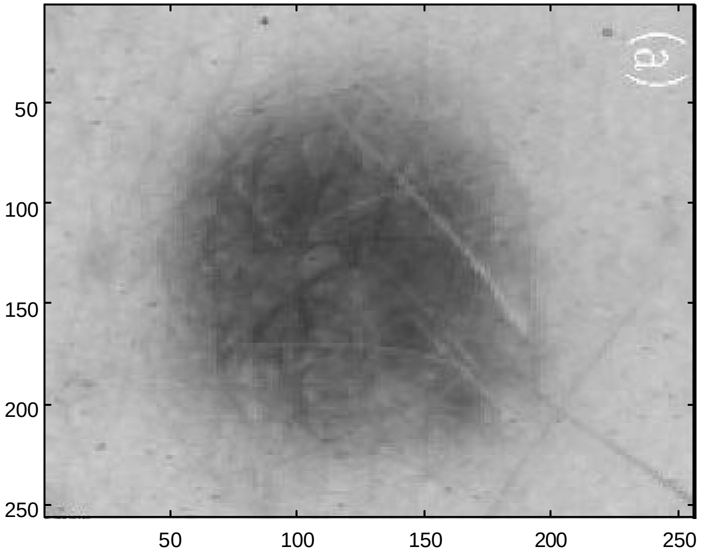


Figure 1 Picture of skin cancer growth

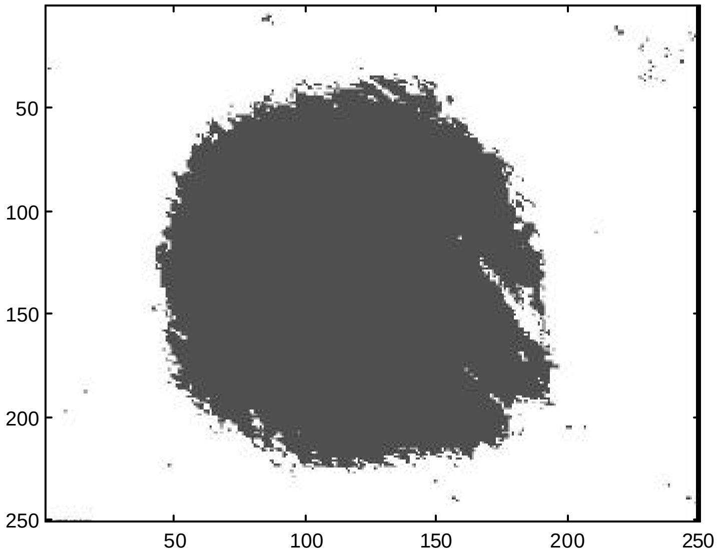


Figure 2 Segmented cancer growth picture

Conclusions

The complex-valued neuron was shown to demonstrate higher computational capability for a large class of problems involving Boolean functions. The complex-valued neuron is able to solve all 16 functions of 2-input Boolean logic, and 245 of the 256 functions of the 3-input Boolean logic. Additionally, the complex valued neuron was shown to be effective at simple computer vision tasks.

Expected benefits of using the complex valued neuron in optical setups include reduced network size, reduced delay when operating in the recall phase, and quicker learning. These benefits will arise because the complex-valued representation will be computationally more powerful than the existing representations. For example, a single complex-valued neuron constructed using the new representation can solve problems that are not linearly separable. Conventional neurons require at least two layers to solve this problem; therefore, ANNs can be constructed with fewer artificial neurons. Although each individual neuron will be more complex, the overall ANN will require less hardware or use fewer mathematical operations to solve existing problems, therefore, speed of operation will be increased and cost will be lowered. These expected benefits are implementation dependent.

The cost of complex-valued neuron is less in all cases than the traditional neuron when implemented optically. Therefore, all the benefits the complex-valued artificial neuron can be obtained without additional cost. Additionally, the complex-valued neuron should be equally superior in those implementations that provide hardware support for complex arithmetic, for example computers with neural-network co-processors based on digital signal processing chips. On those implementations dependent on standard serial computers, the complex-valued neuron will be more cost effective only in those applications where its increased power can offset the requirement for additional neurons.