Complex Valued Signal representation for highly nonlinear processing using Complex Valued interconnected threshold elements with learning capabilities

Abdul A. S. Awwal*[1] and Howard E. Michel*[2]
[1]Computer Engineering/NIF, Lawrence Livermore National Facility; [2]ECE Dept, Umass Dartmouth

Abstract

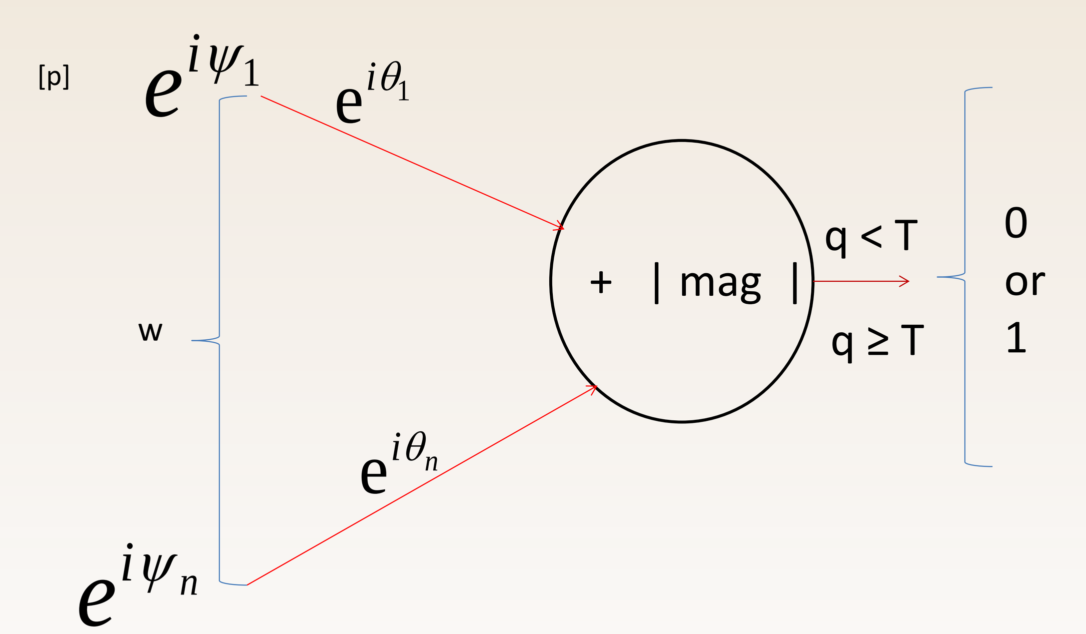
Visual information are inherently 3-D. Complex light signals can be used to capture the 3-D information such as in a hologram. Oppenheim and Lim [1] demonstrated that the phase of a complex Fourier transform can recreate the original scene much better than the amplitude of the Fourier transform alone. It is theorized therefore if information processing can be done in the complex domain using complex valued threshold elements in an interconnected fashion, then more computational capability can be harnessed through such a network. It was shown that using a weighted complex summation and magnitude thresholding, the decision surfaces are naturally curved [2]. Due to its nonlinear decision boundaries, it was shown in 1999 that a single complex valued threshold element is capable of solving the nonlinear XOR problem compared to multilayer perceptrons using real values. In 2003, Nitta also showed similar results [3] using a Complex valued neuron. For three variable Boolean logic, 104 linearly separable problems are solvable by conventional perceptrons. Using Complex valued neural network (CVNN) 135% over that limit of logic operation is achieved without additional logic, neuron stages, or higher order terms such as those required in polynomial logic gates [4]. Although magnitude thresholding was chosen for possible optical implementation [5], computationally one can employ a phase and/or complex thresholding [6].

* When the work was done both authors were with Wright State University, Dayton, Ohio

References

1. A. V. Oppenheim and J. S. Lim, "The importance of phase in signals," IEEE, Proceedings, vol. 69, p. 529-541, 1981.
2. H. E. Michel, and A. A. S. Awwal, "Enhanced Artificial Neural Networks Using Complex-Numbers," Proceedings of the International Joint Conference on Neural Networks. pp. 456-461, 1999.
3. T. Nitta, "Solving the XOR problem and detection of symmetry using a single complex-valued neuron," Neural Networks Letter, Vol. 16, pp. 1101-1105, 2003.
4. H. E. Michel and A. A. S. Awwal, "Artificial neural networks using complex numbers and phase encoded weights, " Applied Optics, Vol. 49, pp. B71-B82, 2010.
5. H. E. Michel, and A. A. S. Awwal, and David Rancour, "Artificial Neural Networks Using Complex Numbers and Phase Encoded Weights--Electronic and Optical Implementations," Proceedings of the International Joint Conference on Neural Networks (IJCNN) 2006, pp. 1213-1218, Vancouver, BC, Canada, July 2006.
6. Aizenberg, N. N., Yu. L. Ivas'kiv, D. A. Pospelov and G. F. Khudyakov, "Multivalued threshold functions. I. Boolean complex-threshold functions and their generalization", Cybernetics and Systems Analysis, vol. 7, No 4, July 1971, pp. 626-635.

A Complex Valued Neuron



Performs complex multiplication, complex addition and magnitude thresholding

Input mapping: from real value to Complex input

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

Mathematics of a Complex Valued Neuron

P = input phase vector
W = weight
q = weighted sum , a = thresholded output

$$\mathbf{p} = \begin{pmatrix} e^{i\psi_1} & e^{i\psi_2} & \dots & e^{i\psi_n} & 1 \end{pmatrix}^{\mathrm{T}}$$

$$\mathbf{w} = \begin{pmatrix} e^{i\theta_1} & e^{i\theta_2} & \dots & e^{i\theta_n} & \lambda_b e^{i\theta_b} \end{pmatrix}$$

$$q = \mathbf{wp} = \sum e^{i(\theta_i+\psi_i)} + \lambda_b e^{i\theta_b}$$

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

The XOR Problem

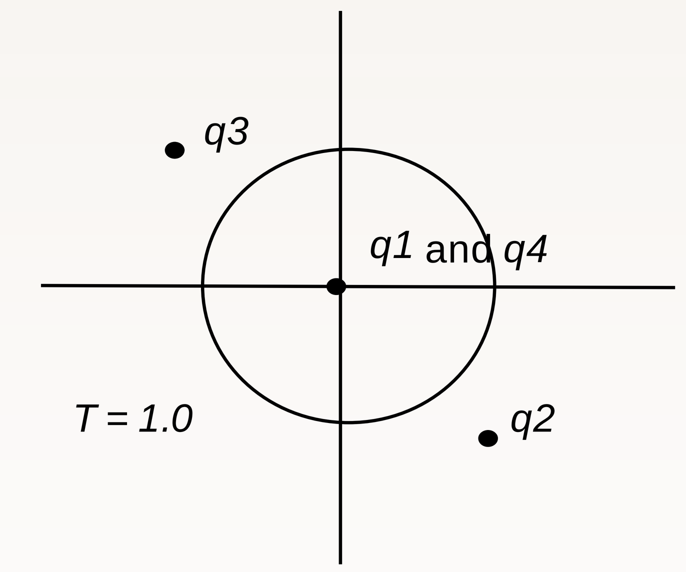
Input		XOR	Intermediate
0	0	0	q_1
0	1	1	q_2
1	0	1	q_3
1	1	0	q_4

Input mapping

$0 \Rightarrow e^{(j\pi/2).0} \Rightarrow 1$

$1 \Rightarrow e^{(j\pi/2).1} \Rightarrow j$

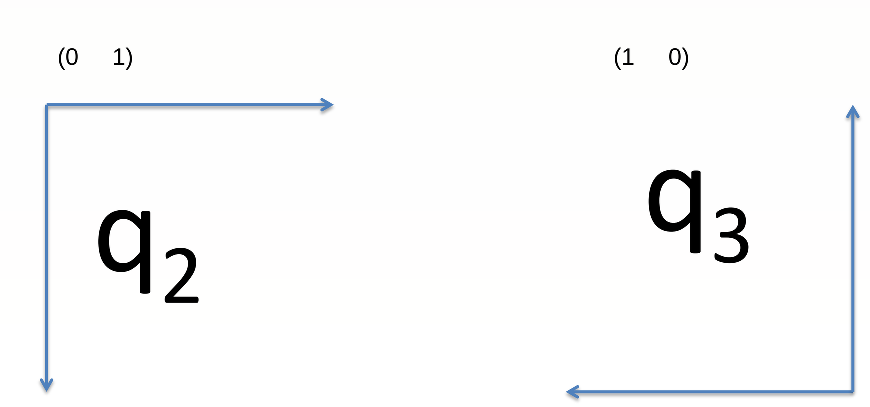
Complex-valued intermediate space for XOR problem



XOR Solution Weights

$$\mathbf{w} = \begin{pmatrix} e^{0} & e^{j\pi} \end{pmatrix}$$

For (0 0), q = 1(1) + 1(-1) = 0
For (1 1), q = j(1) + j(-1) = 0
For (0 1),q = 1.1+$e^{i\pi/2}$.$e^{j\pi}$ = 1 + $e^{i\,3\pi/2}$
For (1 0),q = $e^{i\pi/2}$.1+.$e^{j\pi}$ = $e^{i\pi/2}$ + $e^{i\pi}$



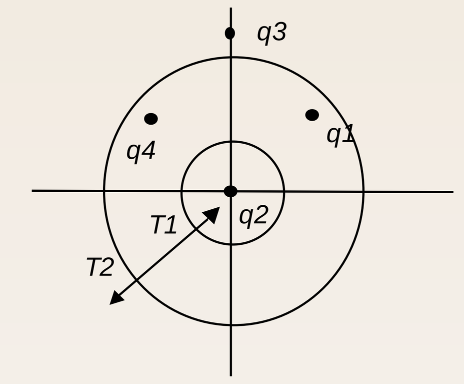
Training and more results

All terms such as bias, threshold and the weights can be trained to learn a specific function

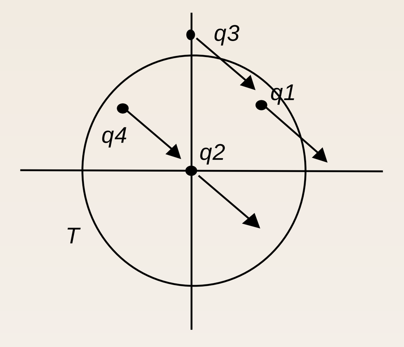
$$b = \lambda_{\mathbf{b}} e^{i\theta_b}$$ **Bias term**

$$\mathbf{w} = \begin{pmatrix} \lambda_{w_1} e^{i\theta_1} & \lambda_{w_2} e^{i\theta_2} & \lambda_{\mathbf{b}} e^{i\theta_b} \end{pmatrix}$$ **Weight with bias term**

$$T_{\text{new}} = T_{\text{old}} - \eta(d-a)T_{\text{old}}$$ **Training threshold**



Effect of changing threshold *T*



Effect of adding *bias* term

$$\lambda_{b\,\text{new}} = \lambda_{b\,\text{old}} + \frac{\eta(d-a)}{\frac{\delta r}{\delta\lambda_b}}$$ **Training bias**

Learning the weights

$$\begin{bmatrix} \theta_{1\text{new}} & \theta_{2\text{new}} \end{bmatrix} = \begin{bmatrix} \theta_{1\text{old}} & \theta_{2\text{old}} \end{bmatrix} + \eta(d-a)\begin{bmatrix} \frac{1}{\frac{\delta r}{\delta\theta_1}} & \frac{1}{\frac{\delta r}{\delta\theta_2}} \end{bmatrix}$$

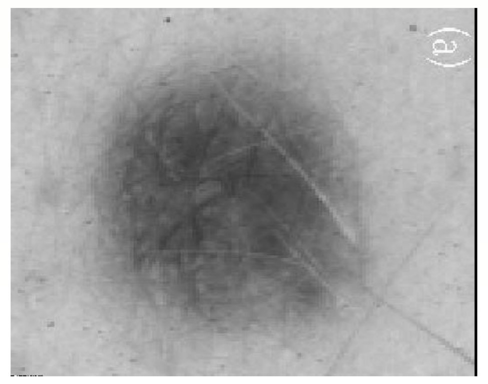
$$\frac{\delta r}{\delta\theta_i} = -2\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\sin(\theta_i+\psi_i-\theta_j-\psi_j) - 2\lambda_b\sum_{i=1}^{n}\sin(\theta_i+\psi_i-\theta_b)$$

Results

Y = $y_1y_2y_3y_4$	θ_1	θ_2	θ_b
0000	-0.3246	1.4096	-1.7286
0001	1.5425	0.1151	-1.9035
0010	-1.0964	1.9306	1.0902
0011	0.0973	2.8959	-3.0582
0100	-2.4710	-0.2429	2.3096
0101	0.6664	-2.0151	0.7127
0110	-2.3024	1.0335	1.7290
0111	-0.6707	-0.4818	2.2412
1000	1.9473	1.9035	0.1437
1001	-2.4268	-3.0646	1.1171
1010	1.3255	-3.1392	0.7240
1011	2.1486	-2.3042	-1.7459
1100	-0.5393	-1.6445	-2.0235
1101	0.6765	-0.4642	-0.6235
1110	1.1095	0.8900	-0.0889
1111	0.2856	0.4488	0.6283

Learned weights, in radians, for 2-input-plus-bias complex-valued perceptron For 16 functions of 2 input Boolean logic

For 3 input Boolean logic 245 out of 256 functions were learned



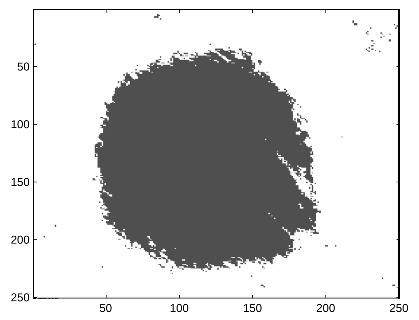


Image segmentation using CVNN

Future: Extend to deep learning

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.