LETTER

A Solution to the 4-bit Parity Problem with a Single Quaternary Neuron

Tohru Nitta

National Institute of Advanced Industrial Science and Technology (AIST),
AIST Tsukuba Central 2, 1-1-1 Umezono, Tsukuba-shi, Ibaraki, 305-8568 Japan
E-mail: tohru-nitta@aist.go.jp

(Submitted on August 20, 2004)

Abstract - This letter will clarify the fundamental properties of a *quaternary neuron* whose weights, threshold values, input and output signals are all quaternions, which is an extension of a usual real-valued neuron to quaternions. The main results of this letter are summarized as follows. A quaternary neuron has an orthogonal decision boundary. The 4-bit parity problem which cannot be solved with a single usual real-valued neuron, can be solved with a single quaternary neuron with the orthogonal decision boundary, resulting in the highest generalization ability.

Keywords - Quaternion, decision boundary, parity problem

1. Introduction

Several neural network models with complex-valued (i.e., two-dimensional) or three-dimensional parameters have been proposed [1, 2, 3, 4, 5, 6] which can deal with complex-valued signals or three-dimensional vectors naturally and demonstrated to have the inherent properties such as the abilities to learn 2D or 3D affine transformations [2, 7, 5, 8, 9, 10]. Particularly, the Complex-BP [2, 5] and the 3DV-BP [3] have been successfully applied to computer vision [11]. We can find some other applications of the complex-valued neural networks to various fields such as optical processing and image processing in the literature [12, 13, 14].

Quaternary neural networks were proposed by Arena and Nitta independently in the mid-1990s [15, 16]. The quaternary neural network is an extension of the classical real-valued neural network to quaternions, whose weights, threshold values, input and output signals are all quaternions where a quaternion is a four-dimensional number and was invented by W. R. Hamilton in 1843 [17]. It is expected that the quaternary neural network can be effectively used in the fields such as robotics and computer vision in which quaternions have been found useful. Actually, it was shown in [18] that the quaternary neural network can solve several problems such as the interpolation of the electric field generated by two charges located in a 3D space, the classification problem of three species of the Iris flower, the chaotic time series prediction, and the attitude control of a rigid body in a 3D space with fewer neurons and connections than the classical real-valued neural network. Isokawa et al. successfully applied a quaternary neural network which calculated a rotation to a color image compression problem [19].

This letter will clarify the fundamental properties of a quaternary neuron. The main results are summarized as follows. The decision boundary of the quaternary neuron consists of four hypersurfaces which intersect orthogonally each other, and divides a decision region into $2^4 (= 16)$ equal sections. The 4-bit parity problem which cannot be solved with a single real-valued neuron, can be solved by a single quaternary neuron with the orthogonal decision boundary, resulting in the highest generalization ability, which reveals a potent computational power of the quaternary neuron.

2. The Quaternary Neuron

There appear to be several approaches for extending the standard neuron to higher dimensions. One approach is to extend the number field, i.e., from real numbers x (1 dimension), to complex numbers $z = x + yi$ (2 dimensions), to quaternions $q = a + bi + cj + dk$ (4 dimensions), to octaves (8 dimensions), to sedenions (16 dimensions), $\cdots$. Another approach is to extend the dimensionality of the weights and threshold values from 1 dimension to n dimensions using n-dimensional real valued vectors. Moreover, the latter approach has two varieties : (a) weights are n-dimensional matrices [3], (b) weights are n-dimensional vectors [4]. In this letter we deal with the quaternary neuron, which is an extension of the real-valued neuron to 4 dimensions in the former approach.

A quaternary neuron is defined as follows [16]. The input signals, weights, thresholds and output signals are all quaternions. The activity A_n (analogous to the real activity in the standard neuron) of neuron n is defined to be:

$$A_n = \sum_m W_{nm} S_m + T_n, \tag{1}$$

where S_m is the quaternary input signal coming from the output of neuron m, W_{nm} is the quaternary weight connecting the neurons m and n, and T_n is the quaternary threshold value of the neuron n. To obtain the quaternary output signal, convert the activity value A_n into its four parts as follows:

$$A_n = x_1 + x_2 i + x_3 j + x_4 k = x, \tag{2}$$

where $i^2 = j^2 = k^2 = -1, ij = -ji = k, jk = -kj = i$ and $ki = -ik = j$. The output signal $1_Q(x)$ is defined to be

$$1_Q(x) \quad = \quad 1_R(x_1) + 1_R(x_2)i + 1_R(x_3)j + 1_R(x_4)k, \tag{3}$$

where 1_R is a real-valued step function defined on $\mathbf{R}$, that is, $1_R(u) = 1$ (if $u \geq 0$), $1_R(u) = 0$ (otherwise) for any $u \in \mathbf{R}$ ($\mathbf{R}$ denotes the set of real numbers).

The multiplication $W_{nm}S_m$ in Eq. (1) should be carefully treated, because the equation $W_{nm}S_m = S_m W_{nm}$ does not hold (the non-commutative property of quaternions on multiplication), which produces two kinds of quaternary neurons: one is called *normal quaternary neuron* which calculates $A_n = \sum_m W_{nm} S_m + T_n$, the other is called *inverse quaternary neuron* which calculates $A_n = \sum_m S_m W_{nm} + T_n$.

3. Orthogonality of Decision Boundary in the Quaternary Neuron

We can find that the decision boundary of a quaternary neuron consists of four hyperplanes which intersect orthogonally each other, and divides a decision region into $2^4 (= 16)$ equal sections as that of a complex-valued neuron case [8]. We show this property below only in the case of a normal quaternary neuron. An inverse quaternary neuron case can be easily shown in a similar manner. The net input U to a normal quaternary neuron with M inputs can be rewritten as:

$$\begin{aligned}
U \quad &= \quad \sum_{l=1}^{M} w_l x_l + \theta \\
&= \quad \sum_{l=1}^{M} (w_l^{(1)} + w_l^{(2)} i + w_l^{(3)} j + w_l^{(4)} k) \cdot (x_l^{(1)} + x_l^{(2)} i + x_l^{(3)} j + x_l^{(4)} k) \\
&\qquad\qquad + (\theta^{(1)} + \theta^{(2)} i + \theta^{(3)} j + \theta^{(4)} k) \\
&= \quad \left\{ [{}^t w^{(1)} \;\; -{}^t w^{(2)} \;\; -{}^t w^{(3)} \;\; -{}^t w^{(4)}] \cdot {}^t[{}^t x^{(1)} \;\; {}^t x^{(2)} \;\; {}^t x^{(3)} \;\; {}^t x^{(4)}] + \theta^{(1)} \right\} \\
&\qquad\qquad + \left\{ [{}^t w^{(2)} \;\; {}^t w^{(1)} \;\; -{}^t w^{(4)} \;\; {}^t w^{(3)}] \cdot {}^t[{}^t x^{(1)} \;\; {}^t x^{(2)} \;\; {}^t x^{(3)} \;\; {}^t x^{(4)}] + \theta^{(2)} \right\} i \\
&\qquad\qquad + \left\{ [{}^t w^{(3)} \;\; {}^t w^{(4)} \;\; {}^t w^{(1)} \;\; -{}^t w^{(2)}] \cdot {}^t[{}^t x^{(1)} \;\; {}^t x^{(2)} \;\; {}^t x^{(3)} \;\; {}^t x^{(4)}] + \theta^{(3)} \right\} j \\
&\qquad\qquad + \left\{ [{}^t w^{(4)} \;\; -{}^t w^{(3)} \;\; {}^t w^{(2)} \;\; {}^t w^{(1)}] \cdot {}^t[{}^t x^{(1)} \;\; {}^t x^{(2)} \;\; {}^t x^{(3)} \;\; {}^t x^{(4)}] + \theta^{(4)} \right\} k,
\end{aligned} \tag{4}$$

where $x^{(s)} = {}^t[x_1^{(s)} \cdots x_M^{(s)}]$ and $w^{(s)} = {}^t[w_1^{(s)} \cdots w_M^{(s)}]$ $(s = 1, 2, 3, 4)$. Thus, the decision boundary of the normal quaternary neuron with M inputs consists of the four equations obtained by letting each term of Eq.(4) be equal to zero. For example,

$$\begin{aligned} & Q(x_1, \cdots, x_M) \\ = \;& [{}^t w^{(1)} \;\; -{}^t w^{(2)} \;\; -{}^t w^{(3)} \;\; -{}^t w^{(4)}] \cdot {}^t[{}^t x^{(1)} \;\; {}^t x^{(2)} \;\; {}^t x^{(3)} \;\; {}^t x^{(4)}] + \theta^{(1)} = 0 \end{aligned} \tag{5}$$

is the decision boundary for the real part of an output of the normal quaternary neuron with M inputs. That is, input signals ${}^t[x_1 \cdots x_M] \in \mathbf{H}^M$ are classified into two decision regions $\{{}^t[x_1 \cdots x_M] \in \mathbf{H}^M \mid Q(x_1, \cdots, x_M) \geq 0\}$ and $\{{}^t[x_1 \cdots x_M] \in \mathbf{H}^M \mid Q(x_1, \cdots, x_M) < 0\}$ by the hyperplane given by Eq.(5) ($\mathbf{H}$ denotes the set of quaternions). We can find that the inner product of the two normal vectors of any two distinct decision boundaries is zero. For example, the inner product of the normal vectors of the decision boundaries for the j-part and k-part is calculated as follows:

$$[{}^t w^{(3)} \;\; {}^t w^{(4)} \;\; {}^t w^{(1)} \;\; -{}^t w^{(2)}] \cdot {}^t[{}^t w^{(4)} \;\; -{}^t w^{(3)} \;\; {}^t w^{(2)} \;\; {}^t w^{(1)}] = 0. \tag{6}$$

Thus, the decision boundary of a normal quaternary neuron consists of four hyperplanes which intersect orthogonally each other.

4. Solving the 4-bit Parity Problem by a Single Quaternary Neuron

We will find a solution to the 4-bit parity problem, using a single normal quaternary neuron with the orthogonal decision boundary with the highest generalization ability. Minsky and Papert clarified the limitations of a single real-valued neuron: in a large number of interesting cases, a single real-valued neuron is incapable of solving the problems [20]. The most difficult problem among them is the parity problem, in which the output required is 1 if the input pattern contains an odd number of 1s and 0 otherwise.

Rumelhart, Hinton and Williams showed that the *3-layered* real-valued neural network (i.e., with one hidden layer) can solve the N-bit parity problem ($N = 2, \cdots, 8$) [21]. As described above, the N-bit parity problem cannot be solved with a single real-valued neuron ($N \geq 2$). Then, it will be proved that the 4-bit parity problem can be solved by a single normal quaternary neuron with the orthogonal decision boundary (i.e., $N = 4$). Rumelhart, Hinton and Williams showed that increasing the number of layers made the computational power of neural networks high. We will show that extending the dimensionality of neural networks to 4 dimensions originates the similar effect on neural networks.

In this connection, the exclusive-or (XOR) problem and the detection of symmetry problem which cannot be solved with a single real-valued neuron [20], can be solved with a single complex-valued neuron with the orthogonal decision boundaries [9, 10].

The input-output mapping in the 4-bit parity problem is shown in Table 1(a). In order to solve the 4-bit parity problem with a normal quaternary neuron, the input-output mapping is encoded as shown in Table 1(b) where the outputs $0, j + k, i + k, i + j, 1 + k, 1 + j, 1 + i$ and $1 + i + j + k$ are interpreted to be 0, and $k, j, i, 1, i + j + k, 1 + j + k, 1 + i + k$ and $1 + i + j$ are interpreted to be 1 of the original 4-bit parity problem (Table 1(a)), respectively. We use a single normal quaternary neuron with only one input and a weight $w = w_1 + w_2 i + w_3 j + w_4 k \in \mathbf{H}$ (we assume that it has no threshold parameters). The decision boundary of the normal quaternary neuron described above consists of the following four hyperplanes which intersect orthogonally each other:

$$[w_1 \;\; -w_2 \;\; -w_3 \;\; -w_4] \cdot {}^t[x_1 \;\; x_2 \;\; x_3 \;\; x_4] = 0, \tag{7}$$
$$[w_2 \;\; w_1 \;\; w_4 \;\; -w_3] \cdot {}^t[x_1 \;\; x_2 \;\; x_3 \;\; x_4] = 0, \tag{8}$$
$$[w_3 \;\; -w_4 \;\; w_1 \;\; w_2] \cdot {}^t[x_1 \;\; x_2 \;\; x_3 \;\; x_4] = 0, \tag{9}$$
$$[w_4 \;\; w_3 \;\; -w_2 \;\; w_1] \cdot {}^t[x_1 \;\; x_2 \;\; x_3 \;\; x_4] = 0 \tag{10}$$

for any input signal $x = x_1 + x_2 i + x_3 j + x_4 k \in \mathbf{H}$. Letting $w_1 = 1$ and $w_2 = w_3 = w_4 = 0$ (i.e., the weight $w = 1$), we can find that the normal quaternary neuron implements the input-output mapping shown in Table 1(b), the decision boundary of which consists of the four orthogonal hyperplanes

$$x_s = 0 \quad (1 \leq s \leq 4) \tag{11}$$

and divides the input space (the decision region) into 2^4 equal sections, and has the highest generalization ability for the 4-bit parity problem.

There exist some neural network models that can solve the N-bit parity problem. The comparison between our result and the previous works for the 4-bit parity problem is shown in Table 2. The number of neurons, the number of parameters, and the number of layers of the normal quaternary neuron are the least. In addition, as described above the generalization ability is the highest. Thus, we conclude that the normal quaternary neuron is the best totally. It should be emphasized here that the number of neurons needed for the normal quaternary neuron is only one (i.e., a single neuron).

Table 1(a). The 4-bit Parity Problem.

Input				Output
x_1	x_2	x_3	x_4	y
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	1	0	0	1
1	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
0	1	1	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Table 1(b). An Encoded 4-bit Parity Problem.

Input				Output
x_1	x_2	x_3	x_4	y
-1	-1	-1	-1	0
-1	-1	-1	1	k
-1	-1	1	- 1	j
-1	1	-1	- 1	i
1	-1	-1	- 1	1
-1	-1	1	1	$j+k$
-1	1	-1	1	$i+k$
-1	1	1	- 1	$i+j$
1	-1	-1	1	$1+k$
1	-1	1	- 1	$1+j$
1	1	-1	- 1	$1+i$
-1	1	1	1	$i+j+k$
1	-1	1	1	$1+j+k$
1	1	-1	1	$1+i+k$
1	1	1	-1	$1+i+j$
1	1	1	1	$1+i+j+k$

Table 2. The Comparison between Our Result and the Previous Works for the 4-bit Parity Problem. *The number of layers* includes an input layer; it is 3 if the network has one hidden layer. *Direct link* means that there are at least one direct link between the input layer and the output layer in the neural network with at least one hidden layer. Note that the number of parameters in Aizenberg's work in the table is the estimated one by the author because Aizenberg et al. solved only the 3, 8 and 9-bit parity problems with a single complex-valued neuron.

	The number of neurons	The number of parameters	The number of layers	Direct link	Activation function
Ours	1	4	2	No	Step function
Setiono [22]	8 or more	16 or more	3	No	Sigmoidal function
Stork and Allen [23]	7	12	3	No	Considerably complicated
Minor [24]	7 or more	9	3	Yes	Sigmoidal function
Lavretsky [25]	3	7	3	Yes	Sigmoidal function
Liu et al. [26]	7 or more	14 or more	3	Yes	Step function
Aizenberg et al. [27]	1	10	2	No	Somewhat special

5. Conclusions

We have found that a single quaternary neuron has the orthogonal decision boundary and can solve the 4-bit parity problem with the highest generalization ability, which suggests that making the dimensionality of neural networks high (from one to four dimensions with the algebraic structure in this letter) is a new directionality for enhancing the ability of neural networks, and that it is worth researching the neural networks with high dimensional parameters. We will apply the quaternary neuron to fields suitable for the orthogonal decision boundary in a future.

References

[1] I. N. Aizenberg, N. N. Aizenberg & J. Vandewalle, *Multi-Valued and Universal Binary Neurons*, Kluwer Academic Publishers, 2000.

[2] T. Nitta & T. Furuya, "A complex back-propagation learning," Transactions of Information Processing Society of Japan, Vol.32, No.10, pp.1319-1329, 1991 (in Japanese).

[3] T. Nitta & H. D. Garis, "A 3D vector version of the back-propagation algorithm," in: Proc. IEEE/INNS International Joint Conference on Neural Networks, Vol.2, pp.511-516, 1992.

[4] T. Nitta, "An extension of the back-propagation algorithm to three dimensions by vector product," in: Proc. IEEE International Conference on Tools with Artificial Intelligence, pp.460-461, 1993.

[5] T. Nitta, "An extension of the back-propagation algorithm to complex numbers," Neural Networks, Vo.10, pp.1392-1415, 1997.

[6] B. Widrow, J. McCool & M. Ball, "The complex LMS algorithm," in: Proc. the IEEE, Vol.63, pp.719-720, 1975.

[7] T. Nitta, "Ability of the 3D vector version of the back-propagation to learn 3D motion," in: Proc. INNS World Congress on Neural Networks, Vol.3, pp.262-267, 1994.

[8] T. Nitta, "An analysis on fundamental structure of complex-valued neuron," Neural Processing Letters, Vol.12, pp.239-246, 2000.

[9] T. Nitta, "Solving the XOR problem and the detection of symmetry using a single complex-valued neuron," Neural Networks, Vol.16, pp.1101-1105, 2003.

[10] T. Nitta, "Orthogonality of decision boundaries in complex-valued neural networks," Neural Computation, Vol.16, No.1, pp.73-97, 2004.

[11] A. Watanabe, N. Yazawa, A. Miyauchi & M. Miyauchi, "A method to interpret 3D motions using neural networks," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E77-A(8), pp.1363-1370, 1994.

[12] A. Hirose & S. Kawata, "Recent progress in coherent lightwave neural systems," in: Proc. International Conference on Neural Information Processing, Vol.3, pp.1074-1078, 2002.

[13] Y. Kuroe, N. Hashimoto & T. Mori, "On energy function for complex-valued neural networks and its applications," in: Proc. International Conference on Neural Information Processing, Vol.3, pp.1079-1083, 2002.

[14] H. Aoki, "A complex-valued neuron to transform gray level images to phase information," in: Proc. International Conference on Neural Information Processing, Vol.3, pp.1084-1088, 2002.

[15] P. Arena, L. Fortuna, L. Occhipinti & M. G. Xibilia, "Neural networks for quaternion valued function approximation," in: Proc. IEEE Int. Symp. on Circuit and Systems, Vol.6, pp.307-310, 1994.

[16] T. Nitta, "A quaternary version of the back-propagation algorithm," in: Proc. IEEE International Conference on Neural Networks, Vol.5, pp.2753-2756, 1995.

[17] H. -D. Ebbinghaus, et al. (Eds.), *Zahlen*, Springer-Verlag, Berlin Heidelderg, 1988 (in German).

[18] P. Arena, L. Fortuna, G. Muscato & M. G. Xibilia, *Neural Networks in Multidimensional Domains, Lecture Notes in Control and Information Sciences*, 234, Springer, 1998.

[19] T. Isokawa, T. Kusakabe, N. Matsui & F. Peter, "Quaternion neural network and its application," in V. Palade, R.J. Howlett and L.C. Jain (Eds.), *LNAI 2774 (KES2003)*, Springer-Verlag, Berlin, pp.318-324, 2003.

[20] M. I. Minsky & S. A. Papert, *Perceptrons*, MIT Press, Cambridge, 1969.

[21] D. E. Rumelhart, G. E. Hinton & R. J. Williams, *Parallel Distributed Processing*, Vol.1, MIT Press, 1986.

[22] R. Setiono, "On the solution of the parity problem by a single hidden layer feedforward neural network," Neurocomputing, Vol.16, pp.225-235, 1997.

[23] D. G. Stork & J. D. Allen, "How to solve the N-bit parity problem with two hidden units," Neural Networks, Vol.5, pp.923-926, 1992.

[24] J. M. Minor, "Parity with two layer feedforward nets," Neural Networks, Vol.6, pp.705-707, 1993.

[25] E. Lavretsky, "On the exact solution of the parity-N problem using ordered neural networks," Neural Networks, Vol.13, pp.643-649, 2000.

[26] D. Liu, M.E. Hohil & S.H. Smith, "N-bit parity neural networks: new solutions based on linear programming," Neurocomputing, Vol.48, pp.477-488, 2002.

[27] N. N. Aizenberg, I. N. Aizenberg & G. A. Krivosheev, "CNN based on universal binary neurons: learning algorithm with error-correction and application to impulsive-noise filtering on gray-scale images," in: Proc. IEEE Int. Workshop on Cellular Neural Networks and their Applications, Seville, Spain, pp.309-314, 1996.

Tohru Nitta received his Ph.D. in information science from University of Tsukuba in 1995, and is currently a Senior Research Scientist in National Institute of Advanced Industrial Science and Technology, Japan and is also an Associate Professor at Department of Mathematics, Graduate School of Science, Osaka University, Japan. His research interests include theoretical issues in computational models such as artificial neural networks, especially complex-valued neural networks. (Home page: http://staff.aist.go.jp/tohru-nitta/)