Research Article

Marat Z. Arslanov*, Zhazira E. Amirgalieva, and Chingiz A. Kenshimov

$N$-bit Parity Neural Networks with minimum number of threshold neurons

DOI 10.1515/eng-2016-0037
Received May 17, 2016; accepted June 24, 2016

Abstract: In this paper ordered neural networks for the $N$-bit parity function containing $\lceil \log_2(N+1) \rceil$ threshold elements are constructed. The minimality of this network is proved. The connection between minimum perceptrons of Gamb for the $N$-bit parity function and one combinatorial problems is established.

Keywords: neural networks, XOR problem, perceptrons of Gamb

1 Introduction

In this article, developing [1], the well known XOR/parity problem is considered. The $N$-bit parity function is a mapping defined on $2^N$ distinct Boolean vectors that indicates whether the sum of the $N$ components of a Boolean vector is odd or even. In [1-11] many solutions of this problem are suggested by various neural networks. Some of these solutions apply complicated activation functions for using neurons. We think, that a choice of different activation functions for solving this problem is non constructive, because there exists the network with only one output neuron with the non monotone activation function

$$f(x) = \begin{cases} 
0 & \text{if } x < 0 \\
0.5 - 0.5 \cos(\pi x) & \text{if } 0 \leq x \leq 2[N/2] + 1 \\
1 & \text{if } x > 2[N/2] + 1,
\end{cases} \tag{1}$$

where $N$ is the number of incoming bits to the output neuron.

The complexity of realizing a Boolean function by some neural network is estimated by the number of neurons. As the threshold element [3] has the most simple structure, the complexity of realizing a Boolean function can be estimated by the number of the threshold elements necessary for its realization.

The threshold element with $n$ incoming bits $x_1, x_2, \ldots, x_n$, $n$ weights $w_1, w_2, \ldots, w_n$, and the threshold $T$ is defined as follows [3]: its output is equal to 1 if $\sum_{i=1}^{n} w_i x_i \geq T$ and 0 otherwise. By using the function

$$\theta(x) = \begin{cases} 
1 & \text{if } x \geq 0 \\
0 & \text{if } x < 0
\end{cases}$$

the output of the threshold element can be written as $\theta(\sum_{i=1}^{n} w_i x_i - T)$.

The XOR problem was solved by threshold neural networks in [4, 5, 8]. In [8] the XOR problem is solved by perceptrons of Gamb, i.e. two-layer neural networks with one output threshold neuron in the second layer, and $m$ threshold neurons in the first layer (see Fig. 1). The thresholds of neurons are in their top part and an input vector $x = (x_1, x_2, \ldots, x_N)$ is presented by one circle.

Analytical formulas are:

$$s_i = \theta \left( \sum_{j=1}^{w} w_{i,j} x_j - T_i \right), \quad i = 1, 2, \ldots, m \tag{2}$$

*Corresponding Author: Marat Z. Arslanov: Institute of information and computation technologies, Almaty 050010, Pushkin str.125, Kazakhstan, E-mail: mzarslanov@hotmail.com
Zhazira E. Amirgalieva: Institute of information and computation technologies, Almaty 050010, Pushkin str.125, Kazakhstan
Chingiz A. Kenshimov: Institute of information and computation technologies, Almaty 050010, Pushkin str.125, Kazakhstan

© 2016 M.Z. Arslanov et al., published by De Gruyter Open.
This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 License.
In [8] the perceptron of Gamb having \( N \) neurons in an intermediate layer for the \( N \)-bit parity function is presented. Moreover, in [8] next estimations are presented

\[
\log_2 N \leq m(N) \leq N \tag{4}
\]

for the minimum number \( m \) of neurons in the first layer of the perceptron of Gamb for the \( N \)-bit parity function. There the problem about exacter estimations of \( m(N) \) is posed. It is possible to show, that for \( N = 2, 3, 4 \) \( m(N) = N \). Nevertheless, the problem about the minimum perceptrons of Gamb for the \( N \)-bit parity function remains open.

In [4, 5] neural networks which generalize perceptrons of Gamb are considered. In these networks an input vector can be directed into an output neuron (Fig. 2)

\[
s_0 = \theta \left( \sum_{i=1}^{m} v_i s_i - T_0 \right) \tag{3}
\]

\[
b_1 = \theta \left( \sum_{i=1}^{N} w_{1,i} x_i - T_1 \right) \tag{7}
\]

\[
b_l = \theta \left( \sum_{i=1}^{N} w_{l,i} x_i + \sum_{j=1}^{l-1} v_{j,l} b_j - T_l \right) , \quad l = 2, 3, \ldots, m \tag{8}
\]

where \( m \) is the number of neurons in the network and the last neuron is an output of the network. In [6] solving the XOR problem is offered for the \( N \)-bit input by the ordered neural network with \( N \) threshold neurons. In Section 2 the ordered neural network with \( \lceil \log_2(N + 1) \rceil \) threshold neurons is offered, that is less, than in [4–6]. It is also shown that there are no ordered neural networks for the \( N \)-bit parity function with smaller number of threshold neurons. In Section 3 some remarks about the minimality problem for the perceptron of Gamb and its connection with one combinatorial problem are given.

In [4, 5] such a neural network for the \( N \)-bit parity function is constructed. It contains \( \lceil N/2 \rceil \) neurons in an interior layer, that twice less upper estimation (4) for the perceptron of Gamb. However, the minimality on number of neurons of this network is not proved. Thus for the architecture (5)-(6) of neural networks there is an open problem about the minimum number of neurons in an intermediate layer for the \( N \)-bit parity function.

We consider solving the XOR problem by the ordered neural network [6]. In the ordered neural network neurons are numbered, and any neuron \( n_1 \) (including input units) can be connected to any other neuron \( n_2 \) (including output units), as long as \( n_1 < n_2 \). Ordered networks generalize feedforward neural networks. In Fig. 3 the structure of the ordered neural network is presented.
2 Ordered neural networks for solving the XOR problem

The construction of the ordered neural network for solving the XOR problem is based on representing the sum of all inputs in the binary notation. It is easy to see, that the value of the N-bit parity function is equal to the latest digit. The structure of such a neural network for the value of the of all inputs in the binary notation. It is easy to see, that

\[ x = (x_1, ..., x_N). \]

Figure 4: Ordered neural network for 15-bit parity function

Analytical formulas are:

\[
\begin{align*}
    b_1 &= \theta \left( \sum_{i=1}^{15} x_i - 7.5 \right), \\
    b_2 &= \theta \left( \sum_{i=1}^{15} x_i - 8b_1 - 3.5 \right), \\
    b_3 &= \theta \left( \sum_{i=1}^{15} x_i - 8b_1 - 4b_2 - 1.5 \right), \\
    b_4 &= \theta \left( \sum_{i=1}^{15} x_i - 8b_1 - 4b_2 - 2b_3 - 0.5 \right).
\end{align*}
\]

It is easy to see, that \( b_1b_2b_3 \) is the binary notation of \( \sum_{i=1}^{N} x_i \). The simple generalization of this example gives the following neural network, which calculates the parity function of N input bits \( x = (x_1, x_2, ..., x_N) \), and contains \( k = \lceil \log_2(N+1) \rceil \) threshold elements:

\[
\begin{align*}
    b_1 &= \theta \left( \sum_{i=1}^{N} x_i - 2^{k-1} + 0.5 \right), \\
    b_l &= \theta \left( \sum_{i=1}^{N} x_i - \sum_{j=1}^{l-1} 2^{k-j}b_j - 2^{k-l} + 0.5 \right), \quad l = 2, 3, ..., k.
\end{align*}
\]

It is easy to see, that \( b_1b_2b_3b_4 \) is the binary notation of \( \sum_{i=1}^{N} x_i \). Let \( c_1c_2\ldots c_k \) be the binary notation of the number \( y = \sum_{i=1}^{N} x_i \). Obviously \( c_1 = 1 \) if and only if \( y \geq 2^{k-1} \), i.e. \( b_1 = c_1 \). Thus, \( y - 2^{k-1} \cdot c_1 \) is less than \( 2^{k-1} \) and it is possible to apply an induction to the binary notation for \( y - 2^{k-1} \cdot c_1 \).

There is a natural question about the existence of ordered neural networks for the N-bit parity function with smaller number of threshold neurons. The further part of this paper is devoted to the negative answer to this question. The idea of the decision is based on the geometrical fact about the structure of a boolean cube \( B^N \) placed in \( R^N \). It turns out, that for every hyperplane in \( R^N \) one of \( [N/2] \)- facets of \( B^N \) lies on one side of a hyperplane.

**Definition 1.** The facet of the dimension \( n \) of a boolean cube

\[ B^N = \{ x = (x_1, x_2, ..., x_N) \mid x_i \in \{0, 1\}, i = 1, 2, ..., N \} \]

or \( n \)-facet is the set of all boolean vectors with \( N - n \) fixed components.

**Lemma 1.** For arbitrary threshold function \( b(x) \) of \( N \) boolean variables \( x = (x_1, x_2, ..., x_N) \) there exists the \( [N/2] \)-facet of a boolean cube \( B^N \) on which this function is constant.

**Proof.** Let the threshold function

\[
b(x) = \begin{cases} 
1, & \text{if } \sum_{i=1}^{N} w_i x_i \geq T, \\
0, & \text{if } \sum_{i=1}^{N} w_i x_i < T
\end{cases}
\]

be given. If some weights \( w_i \) are negative, then by the standard substitution \( x_i = 1 - x_i \) for all such \( i \) and \( x_i = x_i \) for others we receive the threshold function

\[
c(x) = \begin{cases} 
1, & \text{if } \sum_{i=1}^{N} |w_i| x_i \geq T + \sum_{i \in \bar{J}} |w_i|, \\
0, & \text{if } \sum_{i=1}^{N} |w_i| x_i < T + \sum_{i \in \bar{J}} |w_i|
\end{cases}
\]

where \( \bar{J} = \{ i \mid w_i < 0 \} \), such that \( b(x) = c(x') \). So, without loss of generality we can consider, that all \( w_i \geq 0 \), \( i = 1, 2, ..., N \). Let

\[
\sum_{i=1}^{[N/2]} w_i \geq T.
\]

It is easy to see, that on the following facet of the dimension \( N - [N/2] \geq [N/2] \)

\[
F_1 = \{ x \in B^N \mid x_1 = x_2 = ... = x_{[N/2]} = 1 \}
\]

the function \( b(\cdot) \) is equal to 1. If, on the other hand,  

\[
\sum_{i=1}^{[N/2]} w_i < T,
\]
then on the \([N/2]\text{-facet}\)

\[ \{ x \in B^N | x_{[N/2]+1} = x_{[N/2]+2} = \ldots = x_N = 0 \} \quad (17) \]

the function \(b(\cdot)\) is equal to 0. The lemma is proved. \(\square\)

Note that it is impossible to increase the dimension of the facet on which the threshold function is constant. An example is the threshold function \(b(x) = \theta(\sum_{i=1}^{N} x_i - [N/2] - 0.5)\). It is easy to show, that on any facet of the dimension greater than \([N/2]\) this function is not constant.

Now we can prove the minimality of the ordered neural network (10)-(11) on number of threshold neurons.

**Theorem 1.** Let \(N = 2^m\). Then the \(N\)-bit parity function and its negation cannot be presented by an ordered neural network with \(m\) threshold neurons.

**Proof.** We prove the theorem by induction. For \(m = 1\) the theorem asserts the well known fact of threshold logic [3, 8] that the function of "exclusive-or" \(x_1 \oplus x_2\) and its negation \(x_1 \oplus x_2 \oplus 1\) are not threshold functions. Let the theorem be valid for \(m = k\). On the other hand, let for \(m = k + 1\) and \(N = 2^{k+1}\) the \(N\)-bit parity function be presented by the ordered neural network with \(k + 1\) threshold neurons, in the analytical form

\[ b_1 = \theta\left(\sum_{i=1}^{N} w_{i,1}x_i - T_1\right), \quad (18) \]

\[ b_l = \theta\left(\sum_{i=1}^{N} w_{i,l}x_i + \sum_{j=1}^{l-1} v_{j,l}b_j - T_l\right), \quad l = 2, 3, \ldots, k + 1. \quad (19) \]

The contradiction turns out as follows. Consider the first neuron of the given ordered neural network. Its output \(b_1\) depends only on \(N\) input bits, i.e. it is the boolean threshold function \(b_1(x)\) of the boolean vector \(x = (x_1, x_2, \ldots, x_N)\). According to Lemma 1 there exists the facet of the dimension \(N/2 = 2^k\) on which the given function is constant. By the permutation of variables one can achieve that this facet is defined by fixing the variables \(x_{2^k+1}, x_{2^k+2}, \ldots, x_{2^{k+1}}\) or this facet is defined by equations

\[ F = \{ x \in B^N | x_{2^k+1} = \alpha_1, x_{2^k+2} = \alpha_2, \ldots, x_{2^{k+1}} = \alpha_{2^k} \}. \quad (20) \]

If to fix variables \(x_{2^k+1}, x_{2^k+2}, \ldots, x_{2^{k+1}}\) in such a way, the given ordered neural network can be considered as the realization of the boolean function of \(N/2\) variables \(x_1, x_2, \ldots, x_{2^k}\). Clearly, an output of this network is the boolean function \(\bigoplus_{i=1}^{2^k} x_i \oplus \alpha\), where \(\alpha = \bigoplus_{i=1}^{2^k} \alpha_i\). Thus if \(\alpha = 0\) we have ordered neural network for the parity function of \(x_1, x_2, \ldots, x_{2^k}\), or if \(\alpha = 1\) for its negation. It is easy to see, that we can reorganize this network by omitting the first neuron as it is constant and decreasing the thresholds \(T_2, T_3, \ldots, T_{k+1}\) of other neurons by multiplying this constant by appropriate weights. So, we receive the network with \(k\) threshold neurons which calculates, contrary to the assumption of the induction, \(2^k\)-bit parity function or its negation by the formulas:

\[ b_2 = \theta\left(\sum_{i=1}^{N} w_{2,i}x_i - (T_2 - v_{1,2}b_1)\right), \quad (21) \]

\[ b_l = \theta\left(\sum_{i=1}^{N} w_{l,i}x_i + \sum_{j=2}^{l-1} v_{j,l}b_j - (T_l - v_{1,l}b_1)\right), \quad (22) \]

in which the first neuron has an index 2 etc. This contradiction proves the theorem for the \(N\)-bit parity function. Analogously the theorem is proved for the negation of the \(N\)-bit parity function. \(\square\)

From the proved theorem the minimality on number of threshold neurons of the designed ordered neural network (10)-(11) for the \(N\)-bit parity function follows. Indeed, if for some \(N\) there is an ordered neural network for the \(N\)-bit parity function with smaller than \(k = \lfloor \log_2(N + 1) \rfloor\) number of threshold neurons, then, as \(N \geq 2^{k}-1\), from here follows an existence of the ordered neural network with \(k - 1\) threshold neurons for the \(2^{k-1}\)-bit parity function. This contradicts with Theorem 1.

### 3 Some notes about minimum perceptrons of Gamb

Let’s consider the problem of the perceptron of Gamb

\[ s_i = \theta\left(\sum_{j=1}^{N} w_{i,j}x_j - T_i\right), \quad i = 1, 2, \ldots, m \quad (23) \]

\[ s_0 = \theta\left(\sum_{i=1}^{m} v_i s_i - T_0\right) \quad (24) \]

for the \(N\)-bit parity function with minimum number of intermediate threshold neurons, which we denote by \(m(N)\). The upper estimation in (4) is obtained by a construction of the concrete perceptron of Gamb. An example is the fol-
lowing perceptron:

\[ s_i = \theta \left( \sum_{j=1}^{N} (-1)^{i+1} x_j - (i - 0.5)(-1)^{i+1} \right), \quad i = 1, 2, \ldots, N \]

\[ s_0 = \theta \left( \sum_{i=1}^{N} s_i - \left\lfloor \frac{N}{2} \right\rfloor - 0.5 \right). \]

The lower estimation in (4) is accompanied by words "it’s not hard to show, that \( \log_2 N \leq m(N) \). However, the lower estimation easily follows from Theorem 1. The bibliographic searching has shown, that other estimations for \( m(N) \) are absent. We shall connect this problem with the more common combinatorial problem.

Let’s consider the problem of the dissection of all edges of a Boolean cube \( B^N \) located in \( B^N \) by the minimum number of hyperplanes, which do not pass through vertices of \( B^N \). Denote this minimum number of hyperplanes by \( m_1(N) \). Next lemma is valid.

**Lemma 2.** The next inequality is true:

\[ m(N) \geq m_1(N). \]

**Proof.** Let (23)-(24) be the minimum perceptron of Gamb for the \( N \)-bit parity function. The thresholds in it can be taken such, that hyperplanes

\[ \sum_{j=1}^{N} w_{ij} x_j = T_i, \quad i = 1, 2, \ldots, m \]

(27)
do not pass through vertices of \( B^N \). We show, that a set of hyperplanes (27) dissects all edges of \( B^N \). If it is not so, there is an edge in \( B^N \)

\[ ((a_1, a_2, \ldots, a_{i-1}, 1, a_{i+1}, \ldots, a_N)), \]

\[ (a_1, a_2, \ldots, 1 - a_1, a_{i+1}, \ldots, a_N)) \]

which is not dissected by all hyperplanes from (27). Then both boundary points of this edge are on one side from each of hyperplanes (27). This means, that all neurons of the perceptron (23)-(24) are constant on this edge. From (24) it follows that an output neuron \( s_0 \) is constant on this edge. But the parity function is not constant on any edge. This inconsistency proves the lemma.

It is easy to see, that for \( N = 2, 3 \) one have \( m_1(N) = N \), as it is impossible by one line to dissect all 4 edges of a quadrat and by two planes to dissect all 12 edges of a 3-dimensional cube. From here follows, that \( m(N) = N \) with \( N = 2, 3 \). It is possible to show, that \( m(4) = 4 \).

In connection with above the following problems are natural.

1. Whether always \( m(N) = N \)?
2. Are there \( N \), for which \( m(N) > m_1(N) \)?

Similar is the problem of the minimality of neural networks with \( \lceil N/2 \rceil \) intermediate threshold neurons constructed in [4, 5].

However, as it is mentioned in [8], these problems have not simple solution.

**4 Conclusions**

We have offered a simple ordered neural network for the \( N \)-bit parity function, which contains \( \lceil \log_2 (N + 1) \rceil \) threshold elements. For a comparison, the ordered neural network in [6] contains \( N \) threshold gates, and in [4, 5] \( \lceil N/2 \rceil + 1 \) threshold gates. The problems of proving the minimality (on number of threshold neurons) for various architecture of neural networks are still difficult. We have proved the minimality of the offered construction for the \( N \)-bit parity function in the class of ordered neural networks. Some considerations of analogous problem for perceptron of Gamb are undertaken and open problems are formulated.

**References**