N -BIT PARITY NEURAL NETWORKS WITH MINIMUM NUMBER OF THRESHOLD NEURONS

M.Z. Arslanov

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

Key words: Neural Networks, XOR problem, Perceptrons of Gamb

AMS Mathematics Subject Classification: 05XX

1 Introduction

In this article, developing the letter [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 by various neural networks are suggested. 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, as it is possible, e.g. to take the network with only one output neuron, the non monotone activation function of which is equal

$$f(x) = \begin{cases} 0 & if\ x < 0 \\ 0.5 - 0.5\cos(\pi x) & if\ 0 \le x \le 2N/2 + 1 \\ 1 & if\ x > 2N/2 + 1, \end{cases} \quad (1)$$

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

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

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 $x = (x_1, x_2..., x_N)$ is presented by one circle.
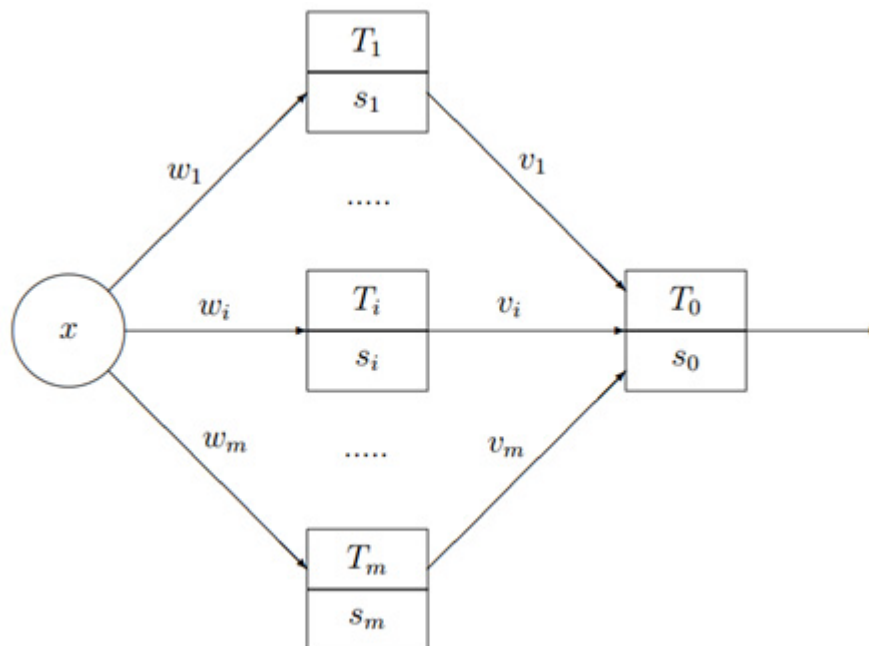

Figure 1. Perceptron of Gamb

Analytical formulas are:

$$s_i = \theta(\sum_{j=1}^{N} w_{i,j} x_j - T_i), i = 1, 2, ..., m \tag{2}$$

$$s_0 = \theta(\sum_{i=1}^{m} v_i s_i - T_0). \tag{3}$$

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 $N = 2, 3, 4 \quad m(N) = N$. Nevertheless, the problem about the minimum on number of neurons 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 direct into an output neuron (Fig.2)
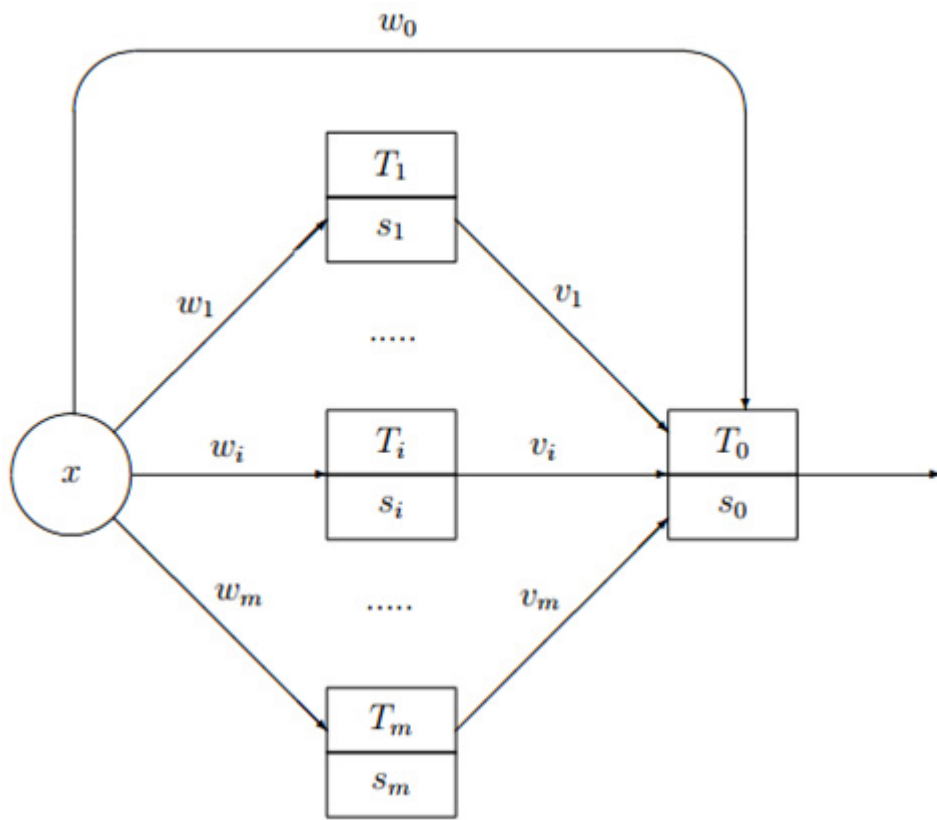
Figure 2. Generalized perceptron of Gamb

Analytical formulas are:

$$s_i = \theta\left(\sum_{j=1}^{N} w_{i,j} x_j - T_i\right), i = 1, 2, ..., m \tag{5}$$

$$s_0 = \theta\left(\sum_{j=1}^{N} w_{0,j} x_j + \sum_{i=1}^{m} v_i s_i - T_0\right). \tag{6}$$

In [4, 5] such a neural network for the N-bit parity function is constructed. It contains $N/2$ neurons in an interior layer, that twice is less then upper estimation (4) for the perceptron of Gamb. However, the minimality on number of neurons of this network is not proved too. 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.
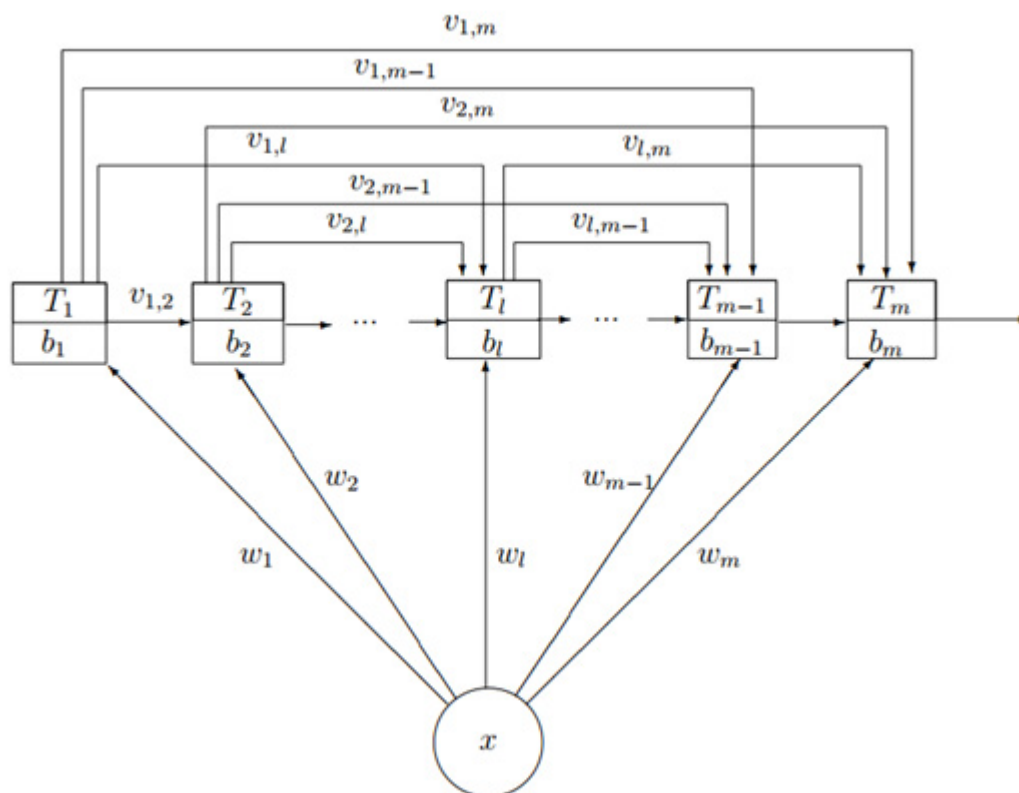
Figure 3. Ordered neural network

Analytical formulas are

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

$$b_l = \theta(\sum_{i=1}^{N} w_{l,i}x_i + \sum_{j=1}^{l-1} v_{j,l}b_j - T_l), l = 2, 3, ..., m,\tag{8}$$

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 $\log_2(N+1)$ threshold neurons is offered, that is less, than in [4, 5, 6]. It is shown also 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.

2   Ordered neural networks for solving the XOR problem

The construction of the ordered neural network for solving the XOR problem is founded on representing the sum of all inputs in the binary notation and the possibility for each

digit in such a representation, beginning from the top digit, to construct the threshold element, output of which is this digit, and inputs are N initial bits and outputs of previous threshold elements, i.e. major digits. 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 $N = 15$, containing 4 threshold elements, is in Fig. 4, where the neuron's threshold is in the top part of the neuron and for convenience all 15 bits are presented by the Boolean vector $x = (x_1, ..., x_{15})$.
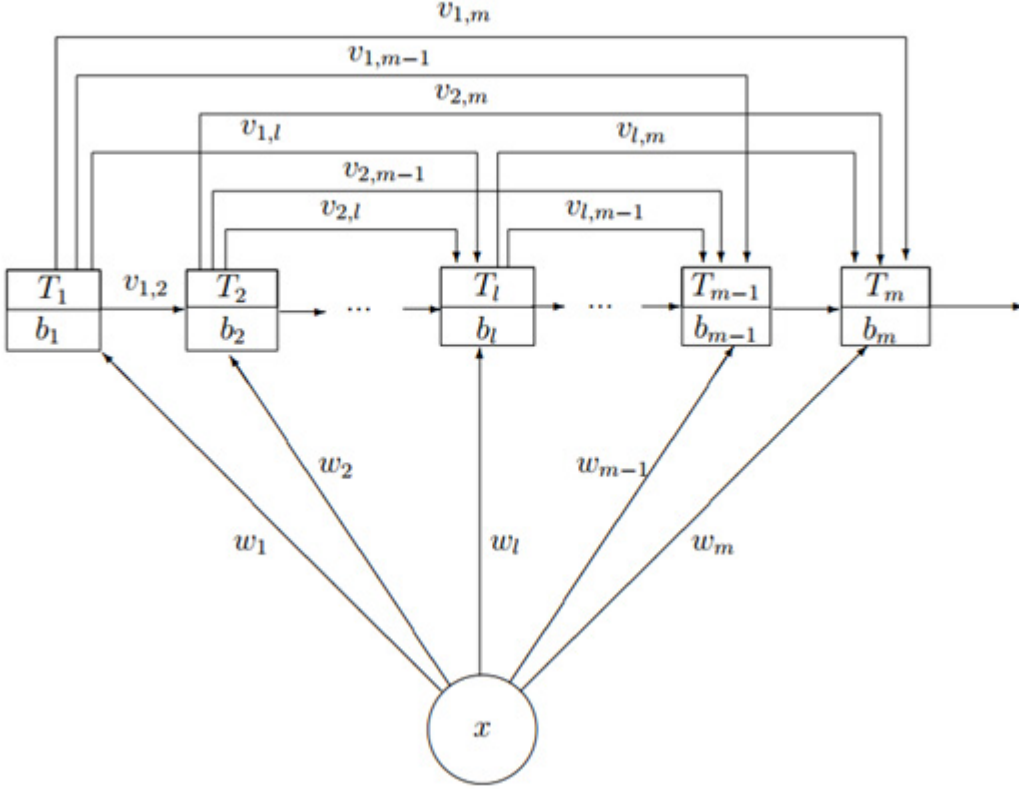

Figure 3. Ordered neural network for 15-bit parity function

Analytical formulas are:

$$b_1 = \theta(\sum_{i=1}^{15} x_i - 7.5), b_2 = \theta(\sum_{i=1}^{15} x_i - 8b_1 - 3.5),$$

$$b_3 = \theta(\sum_{i=1}^{15} x_i - 8b_1 - 4b_2 - 1.5), b_4 = \theta(\sum_{i=1}^{15} x_i - 8b_1 - 4b_2 - 2b_3 - 0.5). \qquad (9)$$

It is easy to see, that $b_1 b_2 b_3$ is the binary notation of $\sum_{i=1}^{7} x_i$. 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 $k = \log_2(N + 1)$ threshold elements:

$$b_1 = \theta(\sum_{i=1}^{N} x_i - 2^{k-1} + 0.5), \qquad (10)$$

$$b_l = \theta(\sum_{i=1}^{N} x_i - \sum_{j=1}^{l-1} 2^{k-j} b_j - 2^{k-l} + 0.5), l = 2, 3, ..., k. \tag{11}$$

It is easy to see, that $b_1 b_2 ... b_k$ is the binary notation of $\sum_{i=1}^{N} x_i$. Really, let $c_1 c_2 ... c_k$, added by zeros at the left if necessary, 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 minimality of constructed neural network, in other words, are there 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 lays on one side from a hyperplane.

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

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

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

Lemma 2. For arbitrary threshold function $b(x)$ from 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

$$b(x) = \begin{cases} 1, & [b]5exif \sum_{i=1}^{n} w_i x_i \geq T, \\ 0, & [b]5exif \sum_{i=1}^{n} w_i x_i < T \end{cases} \tag{12}$$

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

$$c(x') = \begin{cases} 1, & \sum_{i=1}^{n} |w_i| \, x_i' \geq T + \sum_{i \in J} - |w_i| \\ 0, & \sum_{i=1}^{n} |w_i| \, x_i' < T + \sum_{i \in J} - |w_i| \end{cases} \tag{13}$$

where $J^- = \{i | w_i < 0\}$, 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

$$\sum_{i=1}^{N/2} w_i \geq T. \tag{14}$$

It is easy to see, that on the following facet of the dimension $N - N/2 \geq N/2$

$$F_1 = \{x \in B^N | x_1 = x_2 = ... = x_{N/2} = 1\} \tag{15}$$

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

$$\sum_{i=1}^{N/2} w_i < T, \tag{16}$$

then on the $N/2$-facet

$$F_2 = \{x \in B^N | x_{N/2+1} = x_{N/2+2} = ... = x_N = 0\} \tag{17}$$

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

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 3. Let $N = 2^m$. Then the N-bit parity function and its negation can not be submitted by the ordered neural network with m threshold neurons.

Proof. We prove this theorem by an 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 is valid for $m = k$. On the other hand, let for $m = k + 1$ and $N = 2^{k+1}$ the N-bit parity function can be received by the ordered neural network with $k + 1$ threshold neurons, in the analytical form

$$b_1 = \theta(\sum_{i=1}^{N} w_{1,i} x_i - T_1), \tag{18}$$

$$b_l = \theta(\sum_{i=1}^{N} w_{l,i} x_i + \sum_{j=1}^{l-1} v_{j,l} b_j - T_l), l = 2, 3, ..., k + 1. \tag{19}$$

The contradiction turns out as follows. Consider the first neuron of the given ordered neural network. Its output b_1 depends only from N input bits, i.e. it is the boolean threshold function $b_1(x)$ from the boolean vector $x = (x_1, x_2..., x_N)$. 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 $x_{2^k+1}, x_{2^k+2}, ..., x_{2^{k+1}}$ or this facet is defined by equations

$$F = \{x \in B^N | x_{2^k+1} = \alpha_1, x_{2^k+2} = \alpha_2, ..., x_{2^{k+1}} = \alpha_{2^k}\}. \tag{20}$$

If to fix variables $x_{2^k+1}, x_{2^k+2}, ..., x_{2^{k+1}}$ in such a way, the given ordered neural network can be considered as the realization of the boolean function from $N/2$ variables $x_1, x_2, ..., x_{2^k}$. Clearly, an output of this network is the boolean function $\oplus_{i=1}^{2^k} x_i \oplus \alpha$, where $\alpha = \oplus_{i=1}^{2^k} \alpha_i$. Thus if $\alpha = 0$ we have ordered neural network for the parity function from $x_1, x_2..., x_{2^k}$, 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 $T_2, T_3, ..., T_{k+1}$ 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:

$$b_2 = \theta(\sum_{i=1}^{N} w_{2,i} x_i - (T_2 - v_{1,2} b_1)), \tag{21}$$

$$b_l = \theta(\sum_{i=1}^{2^k} w_{l,i}x_i + \sum_{j=2}^{l-1} v_{j,l}b_j - (T_l - v_{1,l}b_1)), l = 3, 4, ..., k+1, \tag{22}$$

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.

From the proved theorem the minimality on number of threshold neurons of designed ordered neural network (10)-(11) for the N-bit parity function follows. Really, if for some N there is an ordered neural network for the N-bit parity function with smaller than $k = \log_2(N + 1)$ 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 synthesis the perceptron of Gamb

$$s_i = \theta(\sum_{j=1}^{N} w_{i,j}x_j - T_i), i = 1, 2, ..., m \tag{23}$$

$$s_0 = \theta(\sum_{i=1}^{m} v_i s_i - T_0) \tag{24}$$

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 following perceptron:

$$s_i = \theta(\sum_{j=1}^{N}(-1)^{i+1}x_j - (i - 0.5)(-1)^{i+1}), i = 1, 2, ..., N \tag{25}$$

$$s_0 = \theta(\sum_{i=1}^{N} s_i - N/2 - 0.5). \tag{26}$$

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 R^N by the minimum number of hyperplanes, which do not pass through vertices of B^N. Let's denote this minimum number of hyperplanes by $m_1(N)$. Next lemma is valid.

Lemma 4. $m(N) \geq m_1(N)$.

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

$$\sum_{j=1}^{N} w_{i,j}x_j = T_i, i = 1, 2, ..., m \tag{27}$$

do not pass through vertices of B^N. Let's show, that a set of hyperplanes (21) dissects all edges of B^N. If it is not so, there is an edge in B^N

$$((\alpha_1, \alpha_2, ..., \alpha_i, \alpha_{i+1}, ..., \alpha_N), (\alpha_1, \alpha_2, ..., 1 - \alpha_i, \alpha_{i+1}, ..., \alpha_N))$$

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 dissect all 4 edges of a guadrate and by two planes dissect all 12 edges of a 3-dimensional cube. From here follows, that $m(N) = N$ with $N = 2, 3$. More hard, but 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 constructed in [4, 5] neural networks with $N/2$ intermediate threshold neurons in a class of neural networks (5)-(6).

However, as it is mentioned in [8], this problems have not simple solution similar to simple constructions of concrete neural networks.

4   Conclusions

We have offered a simple ordered neural network for the N-bit parity function, which contains $\log_2(N+1)$ threshold elements. For a comparison, the ordered neural network in [6] contains N threshold gates, and in [4, 5] $N/2 + 1$ 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

[1] M.Z. Arslanov, D.U. Ashigaliev, E.E. Ismail, *N-bit parity ordered networks*, Neurocomputing, 48 (2002) 1053-1058.

[2] D.A. Brown, *N-bit parity networks, Neural Networks*, 6 (1993) 607-608.

[3] M.L. Dertouzos, Threshold Logic, The M.I.T. Press, Cambridge, Massachusets, 1965.

[4] M.E. Hohil, D. Liu, and S.H. Smith, *Solving the N-bit Parity Problem Using Neural Networks*, Neural Networks 12(9)(November 1999) 1321-1323.

[5] M.E. Hohil, D. Liu, S.H. Smith, *N-bit parity networks: a simple solution, Proceedings of the 32nd Annual Conference on Information Sciences and Systems*, Princeton University, Princeton, NJ, 1998, pp.695-700.

[6] E. Lavretsky,*On the Exact Solution of the Parity - N Problem using ordered Neural Networks*, Neural Networks 13(6)(July 2000) 643-649. .

[7] J.M. Minor,*Parity with two layer feedforward nets*, Neural Networks 6 (1993) 705-707.

[8] M. Minsky, P. Papert P, Perseptrons, *The M.I.T. Press* , Cambridge, Massachusets, 1969.

[9] R. Setiono,*On the Solution of the Parity Problem by a Single Hidden layer Feedforward Neural Network*, Neurocomputing 16(3)(September 1997) 225-235.

[10] D.G. Stork, *N-Bit Parity Networks: A Reply to Brown and Korn*, Neural Networks 6 (1993) 609.

[11] D.G. Stork, J.D. Allen, *How to solve the N-bit parity problem with two hidden units*, Neural Networks 5 (1992) 923-926.

M.Z. Arslanov
Institute of information and computation technologies
Almaty 050010, Pushkin str.125, Kazakhstan
mars@ipic.kz