Neural Networks Computations with DOMINATION Functions

Kordag Mehmet Kilic and **Jehoshua Bruck**
Department of Electrical Engineering, California Institute of Technology, Pasadena 91125, CA, USA

***Abstract*—We study a new representation of neural networks based on DOMINATION functions. Specifically, we show that a threshold function can be computed by its variables connected via an unweighted bipartite graph to a universal gate computing a DOMINATION function. The DOMINATION function consists of fixed weights that are ascending powers of 2. We derive circuit-size upper and lower bounds for circuits with small weights that compute DOMINATION functions. Interestingly, the circuit-size bounds are dependent on the sparsity of the bipartite graph. In particular, functions with sparsity 1 (like the EQUALITY function) can be implemented by small-size constant-weight circuits.**

I. Introduction

The set of n input Boolean functions is the set of functions from $\{0,1\}^n$ to $\{0,1\}$ and these functions are used to model logical systems. Circuit Complexity Theory investigates the idea of computing Boolean functions by networks of simple building blocks called *gates*. The theory questions the upper and lower bounds on the number of gates and number of layers needed to compute a Boolean function by using a family of gates.

An n input MAJORITY function is a Boolean function (i.e. functions from $\{0,1\}^n$ to $\{0,1\}$) that counts the number of 1s and decides 1 if the value exceeds a certain threshold. In circuit complexity theory, limitations and expressive powers of networks of MAJORITY functions are studied and there are several results on this matter [3]. For instance, to make the MAJORITY more expressible, multiple connections of the same input are allowed up to some fan-in constraint [9, 10].

In contrast, we define another function which compares an n bit integer with a fixed integer and decides 1 if the former is greater than or equal to the latter. We call it DOMINATION function because it takes the value 1 depending on the location of the "most significant" 1s in the input vector. Both functions can be realized by a weighted summation and a thresholding element that computes 1 if this summation is non-negative and 0 otherwise. By allowing multiple connections of the same input to different weights, we can similarly generalize the DOMINATION function.

Instead of binary inputs, we will consider input vectors that take values from $\{-n-1, ..., n+1\}$ for some integer n. To illustrate this new form, we will give the definition of the COMPARISON function that computes whether an n bit integer X is greater than or equal to another n bit integer Y.

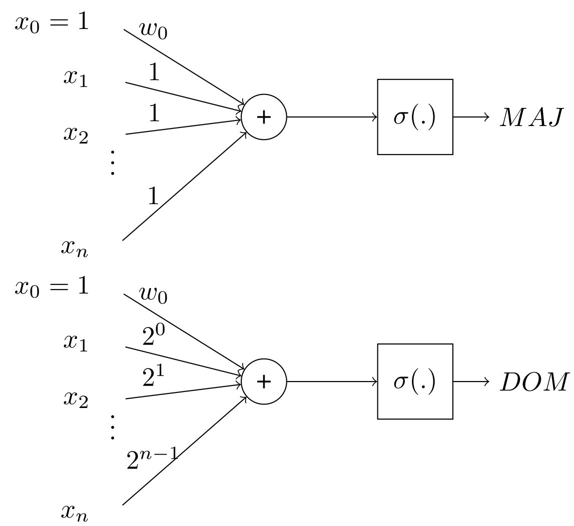

Fig. 1. Explicit gates for n input MAJORITY and DOMINATION functions on $X = (x_1, x_2, ..., x_n)$. w_0 can be chosen arbitrarily to shift the threshold. $\sigma(z) = 1$ if $z \geq 0$ and 0 otherwise. For instance, if $n = 4$ and $X = (0, 1, 0, 1)$ with $w_0 = -3$, $MAJ(X) = 0$ and $DOM(X) = 1$.

$$X \geq Y \Leftrightarrow \sum_{i=1}^{n} 2^{i-1}(x_i - y_i) \geq 0 \tag{1}$$

where $(x_n, ..., x_1)$ and $(y_n, ..., y_1)$ denote the binary expansions of X and Y respectively. We define $X' = X - Y \in \{-1, 0, 1\}^n$ and consequently, a DOMINATION gate with input $X' = X - Y$ can compute COMPARISON function.

All of these functions are examples of a class of Boolean functions called *threshold functions*. For n binary inputs, $X = (x_1, x_2, ..., x_n)$ where $x_i \in \{0,1\}$, we define a *linear threshold function* $f(X)$ by

$$F(X) = w_0 + \sum_{i=1}^{n} w_i x_i \tag{2}$$

$$f(X) = \sigma(F(X)) \tag{3}$$

$$\sigma(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases} \tag{4}$$

We can write any threshold function by using an indicator function as well. Here $-w_0$ is called the *bias* term because it shifts where threshold happens.

$$f(X) = \mathbb{1}\{F(X) \geq 0\} \tag{5}$$
$$= \mathbb{1}\{\sum_{i=1}^{n} w_i x_i \geq -w_0\} \tag{6}$$

We can replace the σ function with any arbitrary function and these general type of gates are called *perceptrons*. In particular, if it is an equality check of the weighted summation to another integer, we call the function *exact* threshold function. It can also be defined as $f(X) = \mathbb{1}\{F(X) = 0\}$. We call the exact version of DOMINATION gate as *exact DOMINATION gate*.

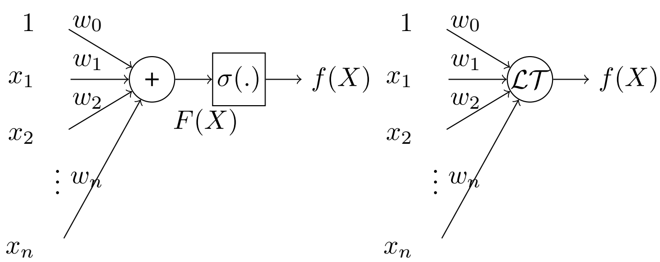

Fig. 2. The diagram for an n input Linear Threshold Gate and an equivalent representation. We use $\mathcal{E}$ instead of $\mathcal{LT}$ if the gate is an exact threshold gate.

In general, to compute an arbitrary (exact) threshold function, a threshold gate requires exponentially large weights in the input size n [1, 4, 11]. A result by Muroga states that any threshold function can be implemented using weights of size $2^{O(n \log n)}$ [14]. An analogous result is obtained by Babai et al. for exact threshold functions [4].

Typically, a threshold function is characterized by the values of its weights. We will reduce this characterization to the input alphabet similar to our approach for COMPARISON. First, we define L to be the least integer satisfying $|w_i| < 2^L$ for all i, i.e. $L = O(n \log n)$. For a weighted summation $F(X)$, let us write the binary expansion of each weight w_i including w_0, and obtain an $L \times (n+1)$ incidence matrix W where w_{ji} is 1 if 2^{j-1} is in the binary expansion of w_i and 0 otherwise. If a weight is negative, we consider the binary expansion of its magnitude and then invert the sign (e.g. $-5 = -(2^2 + 2^0) = -2^2 - 2^0$). Also, we treat $x_0 = 1$ as a fixed input for this representation.

$$\begin{aligned} F(X) &= \sum_{i=0}^{n} w_i x_i = \sum_{i=0}^{n} \Big(\sum_{j=1}^{L} 2^{j-1} w_{ji} \Big) x_i \\ &= \sum_{j=1}^{L} 2^{j-1} \Big(\sum_{i=0}^{n} w_{ji} x_i \Big) = \sum_{j=1}^{L} 2^{j-1} x'_j \end{aligned} \tag{7}$$

In other words, we fix the weights to the powers of two and represent a threshold function by a larger input alphabet where each input can take $n+1$ different values now. If a power of two 2^{j-1} appears in the binary expansion of weights $w_{i_1}, w_{i_2}, ...$, then the corresponding inputs $x_{i_1}, x_{i_2}, ...$ are grouped together in x'_j. In this form, we can compute a threshold function using an $(n+1)$-ary L input DOMINATION gate.

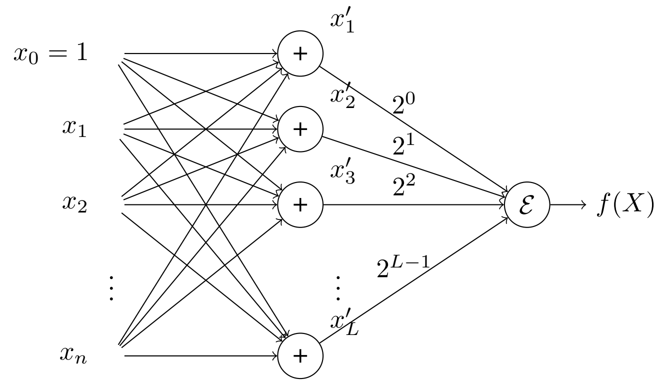

Fig. 3. The construction of an Exact Threshold Gate by an Exact DOMINATION Gate and a layer of adders

To realize this construction in general, we only need to compute WX in a summation layer to obtain X' and then feed it to a DOMINATION gate of higher alphabet. The connectivity graph W has only $\{-1, 0, 1\}$ weights on edges. In this form, we can analyze the connectivity graph and DOMINATION gate separately.

Different threshold functions yield different X' vectors and we want to quantify the incidence relation between X and X'. We define a metric of the sparsity of the graph, $\mathcal{S}$, by counting the maximum possible value each x'_i can take. That is, we define $\mathcal{S} = \max_i |x'_i|$. Since a summation gate is connected to at most $(n+1)$ inputs, it follows that $\mathcal{S} \leq n+1$ and in general, $\mathcal{S} = O(n)$. This $\mathcal{S}$ quantity was also of consideration in various other papers with an almost equivalent definition [2, 12]. For COMPARISON function, it can be seen that $\mathcal{S} = 1$ and this property becomes useful to obtain several important results as it is shown in this paper.

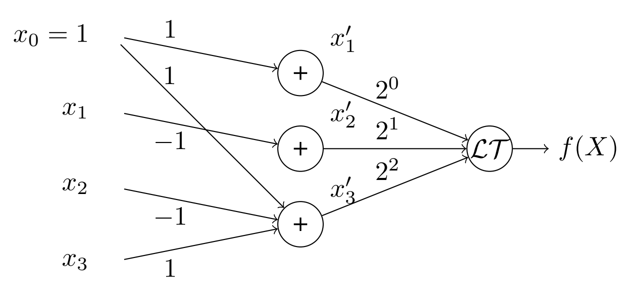

Fig. 4. An example of a construction of a linear threshold function by DOMINATION gate where $f(X) = \mathbb{1}\{-2x_1 - 4x_2 + 4x_3 \geq -5\}$ with $\mathcal{S} = 2$

Because it absorbs all the ambiguity in the unknown weights of a threshold function to a graph structure, this construction provides us a **universal** way to compute threshold functions by different computations of $(n+1)$-ary L input DOMINATION gates. That is to say, if by any means, there is a way to simulate a DOMINATION gate by other types of gates, it can be used to compute any given threshold function. The focus of this paper is to understand how to compute an $(n+1)$-ary L input DOMINATION gate by "simpler" blocks in terms of Circuit Complexity Theory.

The size of weights, circuit size and depth were of interest for neural networks where smaller weight sizes are preferred for better generalization performance and avoidance of gradient explosion problem [5, 7, 8]. Smaller weights and circuit sizes can benefit the efficiency of computation requiring a smaller memory and integer rather than floating point arithmetic. It can be asked whether neural network computations can be done by merely considering the connectivity of inputs to perceptrons as in our framework. Recently, neural networks with binary weights which use a variation of the classical training process were investigated [13]. In general, computation of threshold functions with exponentially large weights in input size n using smaller weights introduces some trade-off at the circuit size and depth, which was treated in many previous papers [9, 12, 17, 15]. However, it is not known if constant weight constructions exist.

Here is a summary of our key contributions:

- An exact threshold function can be universally computed by a depth-2 exact circuit with polynomially large weights in n using $O(\mathcal{S}L^2/\log L)$ many gates.
- If $\mathcal{S} = 1$, this upper bound can be improved to $O(L/\log L)$ and constant weight size constructions exist.
- To compute a $(n+1)$-ary L input DOMINATION function, one needs $\Omega(L/\log nLW)$ many arbitrary perceptrons where W is a weight constraint.
- We can further apply this to EQUALITY function ($\mathbb{1}\{X = Y\}$) to reproduce the same lower bound $\Omega(n/\log n)$ given by [15].

The organization of this paper is as follows. In Section II, we will present Chinese Remainder Theorem (CRT) based constructions for exact DOMINATION functions. In Section III, we will provide a lower bound for DOMINATION gates and by the same technique, present a lower bound for EQUALITY function. In Section IV, for sufficiently large n, we will prove the existence of constructions of depth-2 exact circuits for $\mathcal{S} = 1$ exact threshold functions, which in addition allow $O(1)$ size weights in the first layer. We conclude with some remarks and future directions.

II. Small Weight Construction of DOMINATION functions

We start by introducing the CRT-based constructions given in previous works [2, 10, 12]. For an integer x and modulo base p, we denote the modulo operation by $[x]_p$, which is mapping the integer to the values in $\{0, ..., p-1\}$.

Theorem 1. *Suppose that we are given a function $\mathbb{1}\{F(X) \in C\} = \mathbb{1}\{w_0 + \sum_{i=1}^n w_i x_i \in C\}$ where C is some subset of non-negative integers with largest element is denoted by C_{max}. Let L be the least integer satisfying $|w_i| < 2^L$. Let $C_{max} < p_1 < p_2 < ... < p_r$ be prime numbers and s be the smallest integer such that $\prod_{i=1}^s p_i > (n+1)2^L$. Then,*

1. If $F(X) \in C$, then $[F(X)]_{p_i} \in C$ for all p_i, $i \in \{1, ..., r\}$.

2. If $F(X) \notin C$, then $[F(X)]_{p_i} \in C$ for less than $s|C|$ many primes.

The proof basically follows the idea in [12] using CRT. For exact threshold functions, $C = \{0\}$. Assuming that 2^L dominates $(n+1)$, $s = O(L/\log L)$ by Prime Number Theorem. For separability of $f(X) = 1$ and $f(X) = 0$, we choose $r = s = O(L/\log L)$.

In the circuit construction, we reduce the weights by computing $[w_j]_{p_i}$ for each weight w_j and prime p_i. Let us define $F_i(X) = [w_0]_{p_i} + \sum_{j=1}^n [w_j]_{p_i} x_j$. If we are to use only exact threshold gates in the first layer, we need to compute congruences explicitly, that is, we need to check if $F_i(X) = kp_i$ for some integer k. To do so, the general approach is to use $n+1$ gates together computing $F_i(X) \in \{0, p_i, 2p_i, ..., np_i\}$ because $F_i(X) < (n+1)p_i$.

In the top layer, we similarly use an AND gate to check if $\sum_{k=0}^n \sum_{i=1}^r \mathbb{1}\{F_i(X) = kp_i\} = r$.

Again, primes are of size $O(r \log r)$ by Prime Number Theorem and the maximum weight size becomes $O(np_r) = O(nL)$ as we need to check if $F_i(X) = np_i$ for ith prime. The total number of gates is $(n+1)r + 1 = O(nL/\log L)$. If we take $L = O(n \log n)$, this becomes $O(n^2)$ and this is the best known complexity for depth-2 exact circuit constructions.

Now, we will apply Theorem 1 to the $\mathcal{S}$-ary L input DOMINATION function to obtain an exact circuit construction.

Proposition 1. *The CRT-based depth-2 exact circuit construction applied to $\mathcal{S}$-ary L input exact DOMINATION gate has size complexity $O(\mathcal{S}L^2/\log L)$.*

Proof. For L input DOMINATION gate, recall that $|w_i| < 2^L$ including w_0. Therefore, Theorem 1 gives an upper bound on $r = O(L/\log L)$ similar to arbitrary exact threshold functions.

Since $F(X') < \mathcal{S}Lp_i$, to compute $[F(X')]_{p_i}$ for a prime number p_i, we need to check $F_i(X') \in \{0, p_i, 2p_i, ..., (\mathcal{S}L-1)p_i\}$. Using an AND gate in the top layer, the size complexity becomes $O(\mathcal{S}L^2/\log L)$. □

It turns out that if $\mathcal{S} = O(n)$ and $L = O(n \log n)$, the total size complexity becomes $O(n^3 \log n)$. This is higher than $O(n^2)$ and it seems that there is no use of computing an exact threshold function by reduced weight construction of DOMINATION gate. However, this is not the case for all exact threshold functions. If the graph has sparsity value $\mathcal{S} = 1$, the CRT-based construction of the DOMINATION function can be improved to $O(L/\log L)$ in circuit size. For EQUALITY function, this result is asymptotically efficient with regard to the known lower bounds and our lower bound given in Section III [15].

Proposition 2. *If $\mathcal{S} = 1$, the CRT-based depth-2 exact circuit construction applied to the $\mathcal{S}$-ary L input DOMINATION gate has size complexity $O(L/\log L)$.*

Proof. Basically, we follow Proposition 1 but we will prove that if $\mathcal{S} = 1$, checking $F_i(X') = 0$ suffices.

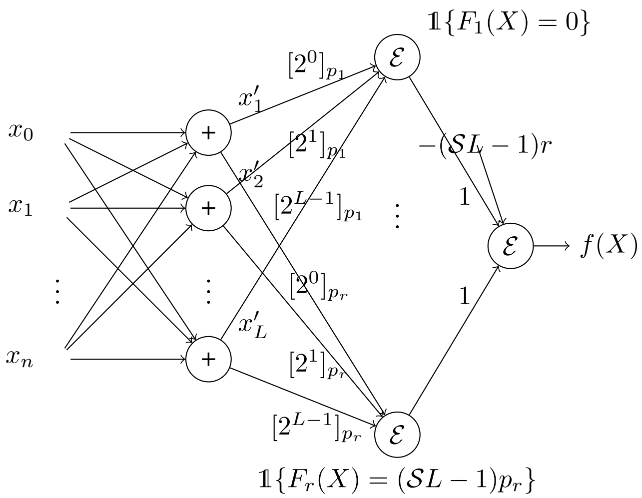

Fig. 5. Depth-2 exact circuit construction of an arbitrary exact threshold gate by using DOMINATION gate and CRT. Notice that DOMINATION gate construction is universal and the function is purely determined by the graph of summation layer.

It follows that if $\mathcal{S} = 1$, the equation $\sum_{j=1}^{L} 2^{j-1} x'_j = 0$ admits unique trivial solution $X' = 0$ (see also Proposition 3). If $X' = 0$, $F_i(X') = \sum_{j=1}^{L} [2^{j-1}]_{p_i} x'_j = 0$ and there is no need to check if $F_i(X') = kp_i$ for $k > 1$. This implies the corresponding upper bound. □

Finally, we remark on the weight sizes of the corresponding constructions. For arbitrary exact threshold functions, the maximum weight size was $np_r = O(nL)$. If $\mathcal{S} > 1$, it is $O(\mathcal{S}L^2)$ (because of the check $F_i(X') = (\mathcal{S}(L+1)-1)p_i$). However, if $\mathcal{S} = 1$, we again see a simplification and the weight of x'_j is $\pm[2^{j-1}]_{p_i}$ for prime p_i. Then, the maximum weight size becomes $O(L)$ by Prime Number Theorem.

III. A Lower Bound for DOMINATION Gates

Let w_b denote the vector of powers of two, $w_b^T = \begin{bmatrix} 2^0 & 2^1 \ldots 2^{L-1} \end{bmatrix}$. Then, we can write $f(X') = \mathbb{1}\{w_b^T x' = 0\}$ for an L input exact DOMINATION gate.

Roughly speaking, we are interested in an equivalent system for $w_b^T x' = 0$ using a matrix $A \in \mathbb{Z}^{m \times L}$, with $Ax' = b$ where entries of A is bounded by some sub-exponential integer. In the circuit constructions, a perceptron computes $z_i = g(A_i X')$ where A_i is the i$^{\text{th}}$ row of A and all gate outputs give some mapping $z = (z_1, ..., z_m)$. For example, $x' = 0$ is mapped to $z = (1, 1, ..., 1)$ if $b = 0$ and we use exact threshold gates in the first layer. Suppose $\mathcal{X}'_1$ is the set of x' where $f(X') = 1$ and $\mathcal{X}'_0$ is the set of x' where $f(X') = 0$ (both form a partition of all available X'). For a valid construction, the sets of output vectors $g(A\mathcal{X}'_1)$ and $g(A\mathcal{X}'_2)$ should not intersect. Conversely, if $g(A\mathcal{X}'_1) \cap g(A\mathcal{X}'_2) \neq \emptyset$, then A cannot be used to construct $f(X)$. This observation leads us to derive a lower bound on the complexity of the circuit size.

To show whether $g(A\mathcal{X}'_1) \cap g(A\mathcal{X}'_2) \neq \emptyset$, we will use the trivial solution $x' = 0$ to the system $w_b^T x' = 0$.

Theorem 2. *To compute an $(n+1)$-ary exact DOMINATION function $f(X')$ with L inputs, the number of $(n+1)$-ary perceptrons with weight constraint W in the circuit is $\Omega(L/\log nLW)$.*

To obtain this lower bound for DOMINATION functions, we apply Siegel's Lemma [16].

Lemma 1 (Siegel's Lemma (modified))**.** *Consider any integer matrix $A \in \mathbb{Z}^{m \times n}$ with $m < n$ and each row-sum of A is bounded by $C \in \mathbb{Z}$ in magnitude. Then, $Ax = 0$ has a non-trivial solution for an integer vector $x \in \mathbb{Z}^n$ such that $\max_j |x_j| \leq C^{\frac{m}{n-m}}$.*

Now, we will give a lemma better suited to prove Theorem 2 (the result of this simple lemma is also envisaged by [6]).

Lemma 2. *Suppose for a matrix $A \in \mathbb{Z}^{m \times n}$, each row-sum is bounded by $C \in \mathbb{Z}$ in magnitude and $m < n \in \mathbb{Z}$. If $m = o(n/\log C)$, then the homogeneous system $Ax = 0$ always admits a non-trivial solution with $\max_j |x_j| = 1$.*

Proof. Since we work in integer domain, it is evident that the upper bound on $\max_j |x_j| \leq 2^{1-\epsilon}$ for any $\epsilon > 0$ implies $\max_j |x_j| = 1$.

Then, by Siegel's Lemma,

$$\max_j |x_i| \leq C^{\frac{m}{n-m}} \leq 2^{1-\epsilon}$$

$$m \leq \frac{n(1-\epsilon)}{1-\epsilon+\log C}$$

In other words, picking any $m = o(n/\log C)$ will guarantee a non-trivial solution to the system $Ax = 0$ such that $\max_j |x_j| = 1$ for all such A matrices. □

Proposition 3. $F(X') = \sum_{i=1}^{L} 2^{i-1} x'_i = 0$ *if and only if $x'_i = 0$ for all $i \in \{1, ..., L\}$ under the assumption $|x'_i| \leq 1$.*

Proof. The converse holds even without the assumption. Under the assumption that $|x'_i| \leq 1$, we can write $x'_i = x_i - y_i$ for $x_i, y_i \in \{0, 1\}$ and use the fact that $X - Y = 0 \Leftrightarrow \sum_{i=1}^{n} 2^{i-1}(x_i - y_i) = 0$. □

Proof of Theorem 2. Since we consider constructions in the form $g(A_i x')$, we can simply take $C = (n+1)LW$ to bound the maximum value each row-sum can take.

On the contrary, suppose there exists a valid construction for an $(n+1)$-ary exact DOMINATION function where we pick the number of gates in the first layer as $o(L/\log nLW)$. Then, Lemma 2 will give a non-trivial solution to the system $Ax' = 0$ with $\max_j |x_j| = 1$, say x^*. By Proposition 3, $w_b^T x^* \neq 0$, therefore, we map at least two inputs with $w_b^T x' = 0$ and $w_b^T x^* \neq 0$ to the same output vector $g(Ax') = g(Ax^*)$, resulting in a contradiction to the existence of a construction. □

This lower bound is not dependent on the depth of the circuits as the argument is based on the information processing of X', in other words, it applies to any depth greater than 2 similar to the bounds given in Roychowdhury et al. [15]. This observation actually gives us a way to generalize the result to threshold DOMINATION functions ($\mathbb{1}\{w_b^T x' \geq 0\}$)

by exploiting the fact that we can simulate any exact threshold function by using 2 linear threshold functions.

Corollary 1. *To compute an $(n+1)$-ary threshold DOMINATION function $f(X')$ with L inputs, the number of $(n+1)$-ary perceptrons with weight constraint W is $\Omega(L/\log nLW)$.*

Proof. Suppose there is a depth-d construction for $\mathbb{1}\{w_b^T x' \geq 0\}$ with $o(L/\log nLW)$ many gates using the entries of a matrix A. We can basically compute $\mathbb{1}\{w_b^T x' = 0\}$ by obtaining a construction for $\mathbb{1}\{w_b^T x' \leq 0\}$. The latter can be done by negating the elements of A. Then, $\mathbb{1}\{w_b^T x' = 0\} = \mathbb{1}\{w_b^T x' \geq 0\} + \mathbb{1}\{w_b^T x' \leq 0\} - 1$ (where no activation function is required). This depth-(d+1) construction computes $\mathbb{1}\{w_b^T x' = 0\}$ with $o(L/\log nLW)$ many gates, resulting in a contradiction. □

Now, we will see how to obtain the lower bound specifically for constructions of EQUALITY function using any perceptron with weight constraint W.

Theorem 3. *To compute the EQUALITY of two unsigned n-bit integers, the number of perceptrons with weight constraint W required in the circuit is $\Omega(n/\log nW)$.*

Consider the EQUALITY function where $F(X) = \sum_{i=1}^{n} 2^{i-1}(x_i - y_i) = 0$. If we apply Theorem 2, we induce a weight mapping from powers of two to weights of size W. Namely, we get $g_j\Big(\sum_{i=1}^{n} a_{ji}(x_i - y_i)\Big)$ for the j$^{\text{th}}$ perceptron. Here, we only account for the constructions in the form $g(AX - AY)$ while we need to consider $g(AX + BY)$.

In fact, this is the main issue regarding the generalization of the lower bound to all exact threshold functions. DOMINATION gate based constructions always use the same weight $\pm a_{ji}$ for binary inputs $x_i, y_i, ...$ inside x'_j while in general, we need the liberty to choose those weights independently.

Proof of Theorem 3. Suppose a construction for EQUALITY gate using perceptrons in the first layer with $m = o(n/\log nW)$ exists (here $C = n|W|$ which is the maximum a weighted summation can take for an n input perceptron with $x_i \in \{0,1\}$) and suppose we have the construction in the form of $g(AX + BY)$. Then, a quadruple (X, Y, Z, T) such that $X \neq Y$ and $Z = T$ with $AX + BY = AZ + BT$ cannot exist. Otherwise, we reach to a contradiction as in the proof of Theorem 2.

Take $X = Z$ and we obtain $B(Y - Z) = 0$. Since B has $m = o(n/\log nW)$ rows, then there is always a non-trivial solution to $B(Y - Z) = 0$ such that $\max_j |Y - Z|_j = 1$ using Lemma 2. This implies the existence of such a (X, Y, Z, T) quadruple. Same arguments apply to matrix A by taking $Y = Z$ to begin with. □

In other words, Theorem 3 states that using different weights for X and Y does not give any benefit on m asymptotically. This is most probably not be true for arbitrary exact threshold functions and the above proof idea does not work in general.

We conjecture that the lower bound $\Omega(L/\log nW)$ applies to the vast majority of the arbitrary exact threshold functions if arbitrary perceptrons with weight constraint W are used in the construction. Furthermore, we believe that this is not tight if we only consider depth-2 exact circuits.

IV. Existence of Constant Weight Constructions for $\mathcal{S} = 1$ Exact Threshold Functions

If we apply the CRT to EQUALITY where $\mathcal{S} = 1$, we can show that the upper bound matches the lower bound with polynomially large weights in n. Next, we will show further reduction of weights is possible.

Theorem 4 (Beck's Theorem for the Converse of Siegel's Lemma [6])**.** *There is a (small) positive constant $c_0 > 0$ with the following property: for every pair $n > m \geq 1$ of positive integers satisfying $n \geq 3m/2$, there exists a matrix A with m rows and n columns, entries of $\{-1, 1\}$ such that for* ***every*** *non-trivial integer solution of the homogeneous system $Ax = 0$, we have*

$$\max_j |x_j| > c_0(\sqrt{n})^{\frac{m}{n-m}} \tag{8}$$

In addition, for large m the overwhelming majority of the $m \times n$ ± 1 matrices A satisfy the theorem. The violators represent an exponentially small $O(2^{-m/2})$ part of the total 2^{mn}.

The theorem proves the fact that Siegel's Lemma is asymptotically sharp and there is a discrepancy between the trivial solution and non-trivial integral solutions to a system $Ax = 0$ in terms of the maximum norm.

Corollary 2. *If $\mathcal{S} = 1$, there are depth-2 exact circuit constructions applied to the S-ary L input DOMINATION gate with size complexity $O(L/\log L)$ and $O(1)$ weights in the first layer.*

The corollary immediately follows by Beck's Theorem because it states that there are $\{-1, 1\}^{m \times L}$ matrices such that every non-trivial integer solution has maximum norm higher than 1 where $m = O(L/\log L)$. The top gate is an AND gate (we ignore the magnitude of the bias term).

In general, for the exact DOMINATION gate $\mathbb{1}\{w_b^T x' = 0\}$, there are many different small norm solutions such as $x' = (-2, 1, 0, ..., 0)$ and we cannot apply Beck's Theorem here. Therefore, we do not know if $\mathbb{1}\{w_b^T x' = 0\}$ admits constant weight perceptron constructions at $O(L/\log L)$ complexity. The answer is affirmative if weights are polynomially large in n considering CRT-based constructions where modulo gates ($g_i(z) = \mathbb{1}\{[z]_{p_i} = 0\}$) are used instead of exact threshold gates. We conjecture that the reduction from polynomial weight sizes to constant weights is true in general.

V. Conclusion and Future Work

We demonstrated a new way of computation for threshold functions using DOMINATION gates. Upper and lower bounds for the small weight circuit constructions of these gates are given. Depending on the sparsity of the graph, existence of the best known constructions in circuit size and constant weights is proved. Extension of these results to arbitrary exact threshold functions is of future interest.

REFERENCES

[1] Noga Alon and Văn H Vũ. "Anti-Hadamard matrices, coin weighing, threshold gates, and indecomposable hypergraphs". In: *Journal of Combinatorial Theory, Series A* 79.1 (1997), pp. 133–160.

[2] Kazuyuki Amano and Akira Maruoka. "Better Simulation of Exponential Threshold Weights by Polynomial Weights". In: *Electronic Colloquium on Computational Complexity (ECCC)* (Nov. 2004).

[3] S Amarel, G Cooke, and Robert O Winder. "Majority gate networks". In: *IEEE Transactions on Electronic Computers* 1 (1964), pp. 4–13.

[4] László Babai, Kristoffer Arnsfelt Hansen, Vladimir V Podolskii, and Xiaoming Sun. "Weights of exact threshold functions". In: *International Symposium on Mathematical Foundations of Computer Science*. Springer. 2010, pp. 66–77.

[5] Peter L Bartlett. "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network". In: *IEEE transactions on Information Theory* 44.2 (1998), pp. 525–536.

[6] József Beck. "Siegel's Lemma is sharp". In: *A journey through discrete mathematics*. Springer, 2017, pp. 165–206.

[7] Monica Bianchini and Franco Scarselli. "On the complexity of neural network classifiers: A comparison between shallow and deep architectures". In: *IEEE transactions on neural networks and learning systems* 25.8 (2014), pp. 1553–1565.

[8] Olivier Delalleau and Yoshua Bengio. "Shallow vs. deep sum-product networks". In: *Advances in neural information processing systems*. 2011, pp. 666–674.

[9] Mikael Goldmann and Marek Karpinski. "Simulating threshold circuits by majority circuits". In: *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*. 1993, pp. 551–560.

[10] Kristoffer Arnsfelt Hansen and Vladimir V Podolskii. "Exact threshold circuits". In: *2010 IEEE 25th Annual Conference on Computational Complexity*. IEEE. 2010, pp. 270–279.

[11] Johan Håstad. "On the size of weights for threshold gates". In: *SIAM Journal on Discrete Mathematics* 7.3 (1994), pp. 484–492.

[12] Thomas Hofmeister. "A note on the simulation of exponential threshold weights". In: *International Computing and Combinatorics Conference*. Springer. 1996, pp. 136–141.

[13] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. "Binarized neural networks". In: *Advances in neural information processing systems* 29 (2016), pp. 4107–4115.

[14] Saburo Muroga. *Threshold logic and its applications*. 1971.

[15] Vwani P Roychowdhury, Alon Orlitsky, and Kai-Yeung Siu. "Lower bounds on threshold and related circuits via communication complexity". In: *IEEE Transactions on Information Theory* 40.2 (1994), pp. 467–474.

[16] Carl L Siegel. "Über einige anwendungen diophantischer approximationen". In: *On Some Applications of Diophantine Approximations*. Springer, 2014, pp. 81–138.

[17] Kai-Yeung Siu and Jehoshua Bruck. "On the power of threshold circuits with small weights". In: *SIAM Journal on Discrete Mathematics* 4.3 (1991), pp. 423–435.