On Threshold Circuits for Parity

Ramamohan Paturi and Michael E. Saks*
Department of Computer Science and Engineering
University of California, San Diego; La Jolla, Ca 92093

Abstract

Motivated by the problem of understanding the limitations of neural networks for representing Boolean functions, we consider size–depth trade–offs for threshold circuits that compute the parity function. We give an almost optimal lower bound on the number of edges of any depth–2 threshold circuit that computes the parity function with polynomially bounded weights. The main technique we use in this proof, which is based upon the theory of rational approximation, appears to be a potentially useful technique for the analysis of such networks. We conjecture that there are no linear size bounded depth threshold circuits for computing parity.

*Research supported in part by NSF grant CCR-8911388

1 Introduction

A basic paradigm of the connectionist approach to learning is that of training a neural network (consisting of weighted threshold gates) by incrementally adjusting the weights in response to concept examples presented to the network [15,8]. Theoretical and empirical evidence suggest that the rate of convergence of such training procedures depends on both the number of edges and the depth of the underlying network [2,8,6]. Thus in selecting an architecture that is appropriate for learning a particular class $\mathcal{F}$ of functions, we have the following criteria:

1. The architecture should be able to compute each function in $\mathcal{F}$ for a proper choice of weights.
2. The architecture should have a minimal number of edges.
3. The depth of this architecture should be minimal.

Unfortunately, there are virtually no analytical tools available for evaluating architectures based on these criteria. Indeed with few notable exceptions [10,9,4,13,7,14,5,16,2], very little is known about the limitations of threshold circuits for representing specific functions. As a step in this direction, it seems worthwhile to consider the trade-offs between the number of edges and the depth of threshold circuits. A natural place to begin such an investigation is with the *parity function* which has served as a prototypical example for a variety of investigations in complexity theory and neural network learning. It is well-known that the parity function on n variables can be computed by a depth–2 threshold circuit with a quadratic number of edges. It is not hard to construct such a circuit with a linear number of edges in depth $O(\log\log n)$ and in fact there is a construction that provides a transition between these extremes (see section 2).

Our starting point was an attempt to show that the trade–off exhibited by this construction is best possible. The result we present in this paper is an almost optimal lower bound $\Omega(n^2/\log^2 n)$ on the number of edges for computing parity function with depth–2 threshold circuits whose weights are bounded by a polynomial in n. (Recently, we have extended the techniques to obtain a $\Omega(n^{1.2-\epsilon})$ lower bound for depth–3 threshold circuits; details will be given in the full paper.) While this result falls short of our goal, its proof involves an interesting application of *rational approximation theory* to complexity theory. The technique holds promise towards a complete answer to this trade-off question and also seems to be a natural and a potentially useful approach to the analysis of threshold circuits in general. Such an approach might also be relevant to the solution of the question whether TC^0 (the class of functions computable by bounded depth and polynomial size threshold circuits) is equal to NC^1 (the class of functions computable by fan–in 2 log–depth Boolean circuits).

2 Threshold Circuits for Parity: Upper Bounds

A threshold gate with n incoming edges is an n-tuple $\vec{w} \in \mathbf{R}^n$. The function $h : \{0,1\}^n \to \{0,1\}$ computed by such a gate on input $\vec{b} \in \{0,1\}^n$ is given by $\mathrm{sgn}(\sum_{i=1}^n w_i b_i)$ where sgn : $\mathbf{R} - \{0\} \to \{0,1\}$ is defined as

$$\mathrm{sgn}(\alpha) = \begin{cases} 1 & \text{if } \alpha > 0 \\ 0 & \text{otherwise} \end{cases}$$

We assume, without loss of generality, that all our gates have the property that for no $\vec{b} \in \{0,1\}^n$, $\sum_{i=1}^n w_i b_i = 0$.

A threshold circuit T on n inputs $(b_1, b_2, \ldots, b_n)$ is a directed acyclic graph with a designated node (output) and exactly $n+1$ *source* nodes, one for each variable and one for the constant 1. Also, each edge is labelled by an *integer*. The semantics of such a circuit is given by interpreting each of the non–source nodes as a threshold gate on its incoming edges and by assigning, in the obvious way, a function of the type $\{0,1\}^n \to \{0,1\}$ to each non–source node as the function computed by that node. The function $f_T : \{0,1\}^n \to \{0,1\}$ computed by T is the function assigned to its designated output node.

The *node complexity* of T is defined as the number of non–source nodes of T. The *edge complexity* of T is defined as the number of edges in T.

We define the *level* of each node in the following way: The level of each source node is 0. The level of any other node i is 1 more than the maximum level of its immediate predecessors. The *depth* of T is the level of the output node.

The node (edge) complexity $N_f(n)$ $(E_f(n))$ of a Boolean function f on n inputs is the node (edge) complexity of the minimal node (edge) threshold circuit that computes f.

For convenience, we assume that all our circuits are layered in that nodes at level i for $i \geq 1$ receive inputs only from level $i-1$. Such a restriction would increase the node and the edge complexities by at most a factor of d, the depth of the circuit.

A Boolean vector $\vec{b}$ is *even* if $\sum_{i=1}^{n} b_i \equiv 0 \bmod 2$ and is *odd* otherwise. The *odd parity* function p_o on a sequence $\vec{b} = (b_1, b_2, \ldots, b_n)$ of Boolean variables is defined to be the characteristic function of the set of odd vectors. The *even parity* function p_e is defined by $p_e(\vec{b}) = 1 - p_o(\vec{b})$. Both even and odd parity functions are referred to simply as *parity* functions.

We now consider the problem of constructing threshold circuits for parity with minimal number of edges. The best construction that has been achieved for depth–2 threshold circuits uses $n^2/2 + O(n)$ edges [10]: At the first level, we have $\lfloor \frac{n}{2} \rfloor$ gates with the i-th gate G_i (which also denotes the output of the gate i) computing the function $\mathrm{sgn}(\sum_{j=1}^{n} 2b_j - 4i + 1)$ for $1 \leq i \leq \lfloor \frac{n}{2} \rfloor$. In addition, we have $n+1$ 'trivial' gates at the first–level that will transmit the inputs $b_1, b_2, \ldots, b_n$ and the constant 1 to the gate at the second–level. The output gate at the second level computes $\mathrm{sgn}(\sum_{j=1}^{\lfloor \frac{n}{2} \rfloor} 4G_i - 2\sum_{i=1}^{n} b_i + 1)$. Note that the argument of the sign function is positive if the input $\vec{b}$ is odd and negative if the input $\vec{b}$ is even.

For higher depth, the above can be used as the basis in a recursive construction. To construct a circuit of even depth d, we partition the inputs into $\sqrt{n}$ sets of size $\sqrt{n}$. We recursively construct depth $d-2$ circuits to compute the parity of each of the sets and combine the outputs using a depth–2 circuit. For $d = o(\log \log n)$, the number of edges at the lowest level of recursion is $O(n^{1+1/2^{d/2}})$ which dominates the sum of all the edges at other levels. This is a special case of a construction by Beame and others ([3]) which, for any symmetric function, yields a similar size–depth trade–off. In the case of the parity function, by a more careful construction, we get:

Theorem 1 *For $d \geq 2$ and $d = O(\log \log n)$, there exists a depth d threshold circuit that computes a parity function having $O(n^{1+1/\Theta(\phi^d)})$ edges and $O(n)$ nodes where $\phi = (1+\sqrt{5})/2$. (All the weights in these circuits are $O(\log n)$ bits long.)*

In particular, we get an $O(\log \log n)$ depth threshold circuit for computing the parity function with node and edge complexities $O(n)$.

3 Threshold Circuits for Parity: Lower Bounds

Our inability to improve the construction of the previous section suggests the following conjecture:

Conjecture 1 *There exists a constant $c > 1$ such that any depth d circuit that computes parity requires $\Omega(n^{1+1/c^d})$ edges.*

This conjecture, if true, would imply that any depth $d = o(\log \log n)$ circuit that computes parity has a superlinear number of edges. In the remainder of this paper, we prove a nearly *quadratic* lower bound on the edge complexity of depth-2 circuits for parity.

3.1 Properties of Depth–2 Parity Circuits

We first establish a relationship between the node and the edge complexities of depth–2 threshold circuits that compute parity. Let p be any parity function on n inputs.

Lemma 1 *For depth–2 threshold circuits,* $(2N_p(n) + nN_p(n-1) - n - 2)/2 \leq E_p(n) \leq (n+2)N_p(n) - n - 2$.

Proof: The upper bound follows easily from elementary graph–theoretic considerations. For the lower bound, we show that the outdegree of each input variable b_i is at least $N_p(n-1)/2$. Let T be a depth–2 threshold circuit that computes a parity function on n inputs. Let $out_T(i)$ be the outdegree of the source labelled by b_i. The idea is that given a depth–2 threshold circuit for computing a parity function on n inputs, we can construct a threshold circuit T' that computes a parity function on $n-1$ inputs whose node complexity is bounded from above by $2out_T(i) + 1$.

T' is constructed from T as follows. Consider the output node v of T. Let T_1 be the circuit obtained from T by merging the source node labelled b_i with the source labelled 1, i.e., assigning the value 1 to the input b_i. Let T_0' be the circuit obtained by deleting the source node labelled b_i together with its incident edges. We then obtain T_0 from T_0' by negating the weights of all the incoming edges of the output node of T_0'. Now, T' is obtained by coalescing the output nodes of T_0 and T_1. Since T_0 and T_1 compute the same parity function on $n-1$ inputs, it is not difficult to see that T' computes a parity function on $n-1$. Notice that all the 1–level nodes of T that do not receive an incoming edge from b_i in T appear as identical pairs in T' except the weights on their outgoing edges have different signs. Therefore, the contribution of each pair to the output node of T' is always 0. Hence we can remove all such nodes from T'. This shows that the node complexity of T' is bounded from above by $2out_T(i) + 1$. From this, we get the lower bound $(N_p(n-1) - 1)/2$ on the outdegree of each non–constant source node which in turn gives the lower bound on $E_p(n)$.

A Geometric View: Each gate in a threshold circuit can be interpreted as an affine hyperplane. With this interpretation, it is not hard to prove:

Lemma 2 *In a depth–2 circuit for parity, the set of hyperplanes associated with the nodes at level 1 must intersect every edge of the n–dimensional unit hypercube.*

This observation suggests the following problem.

Problem 1 *What is the minimal number of hyperplanes required to intersect all the edges of the unit hypercube?*

This problem appears in [12,1] and it was conjectured that this number is n. But, an unpublished counterexample by Paterson is mentioned in [12]. By lemma 1, any $f(n)$ lower bound on the number of hyperplanes would imply an $\Omega(nf(n))$ lower bound on the edge complexity for computing parity with a depth–2 circuit.

In the construction of the previous section, observe that the circuits are *restricted* in the sense that the weights (except the ones that correspond to constant input 1) of all level 1 gates are nonnegative. Therefore the associated hyperplanes have nonnegative coefficients. It is not hard to see that at least n such hyperplanes are needed to cut all of the edges of the hypercube (no two of the n edges $(0^j1^{n-j}, 0^{j+1}1^{n-j-1})$ can be cut by the same nonnegative hyperplane) and thus we have:

Theorem 2 *Any restricted depth–2 threshold circuit that computes parity has edge complexity of* $n^2/2 + \Omega(n)$.

In the case that the weights are not restricted, the best lower bound known on the number of hyperplanes needed to cut all edges of the hypercube is $\Omega(\sqrt{n})$ which gives an $\Omega(n^{3/2})$ bound on the edge complexity of parity with depth–2 circuits. We believe that this can be significantly improved but have been unable to do so this technique. Instead, the nearly quadratic promised in the introduction is based on an analytic approach, which uses approximation by rational functions together with a degree argument.

3.2 Multilinear Polynomials and Parity

Let $f(x_1, x_2, \ldots, x_n)$ be a real multilinear polynomial in $x_1, x_2, \ldots, x_n$. $f(x_1, x_2, \ldots, x_n)$ has the following general form:

$$f(x_1, x_2, \ldots, x_n) = \sum_{S \subseteq [n]} \alpha_S \Pi_{i \in S} x_i$$

where $[n] = \{1, 2, \ldots, n\}$. The degree of f is defined by

$$\deg(f) = \max_{S \subseteq [n]} \{|S| : \alpha_S \neq 0\}.$$

We say that a Boolean function $g : \{0,1\}^n \rightarrow \{0,1\}$ is *computable* by a multilinear real polynomial $f(x_1, x_2, \ldots, x_n)$ if, for all $\vec{b} \in \{0,1\}^n$, $g(\vec{b}) = \mathrm{sgn}(f(\vec{b}))$.

Fact 1 *Every Boolean function of the type* $\{0,1\}^n \rightarrow \{0,1\}$ *is computable by a multilinear real polynomial of degree at most* n.

Lemma 3 *If a multilinear real polynomial* $f(x_1, x_2, \ldots, x_n)$ *computes a parity function, then* $\deg(f) = n$.

Proof: We use induction on n to prove that the $\deg(f)$ is at least n. Equality follows from fact 1.

Clearly, the statement is true for $n = 1$. For some $n \geq 2$, assume that $f(x_1, x_2, \ldots, x_n)$ computes the function p_e on n variables. (The other case can be handled similarly.) We can write f as

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) &= x_1 f_1(x_2, \ldots, x_n) \quad (1) \\ &\quad + f_0(x_2, \ldots, x_n) \quad (2) \end{aligned}$$

such that $f_0(x_2, x_3, \ldots, x_n)$ computes p_e on $n - 1$ variables and $f_1(x_2, x_3, \ldots, x_n) + f_0(x_2, x_3, \ldots, x_n)$ computes p_o on $n - 1$ variables.

It follows that $f_1(x_2, x_3, \ldots, x_n)$ computes p_o on $n-1$ variables. Observe that the degree of f_1 is one less than that of f. By induction hypothesis, we have that $\deg(f_1) \geq n - 1$. Hence, $\deg(f) \geq n$.

Remark: This lemma can be used to give a simpler proof of the result of Alon, Bergmann, Coppersmith and Odlyzko [1], concerning the number of Hamming spheres required to cover all the vertices of the n-dimensional hypercube. More precisely, they considered the following problem due to Knuth. Let $K(n, d)$ denote the minimal k for which there exist ± 1 vectors $v_i, v_2, \ldots, v_k$ of dimension n such that for any ± 1 vector w of dimension n, there is an i, $1 \leq i \leq k$, such that $|v_i.w| \leq d$, where $v.w$ denotes the inner product of two vectors. Knuth's construction shows that $K(n, d) \leq \lceil n/(d+1) \rceil$ for $n \equiv d \pmod 2$. Alon and others proved a matching lower bound using elementary linear algebra. This lower bound can be obtained from lemma 3, by noting that a cover of the hypercube by h Hamming spheres of radius at most d can be used to obtain a polynomial of degree at most $h(d+1)$ that computes parity for all vertices of the hypercube. (We omit the details from this abstract).

3.3 Rational Approximation and Threshold Gates

In this section, we consider the following problem: $|x|$ is defined on the closed interval $[-1,1]$ as the absolute value of x. We wish to approximate $|x|$ in the interval $[-1,1]$ by a rational function $R_k(x)$ in x of degree k where the error of the approximation $E_{R_k(x)}(|x|)$ is measured by

$$E_{R_k(x)}(|x|) = \sup_{x\in[-1,1]} |R_k(x) - |x||.$$

The degree of a rational function is the maximum of the degrees of its numerator and denominator polynomials.

We have the following fundamental results of Newman [11] regarding the approximation of $|x|$. Let $\xi = e^{-1/\sqrt{k}}$, and

$$p(x) = \Pi_{i=0}^{k-1}(x+\xi^i),$$

$$R_{k+1}(x) = x\frac{p(x)-p(-x)}{p(x)+p(-x)}$$

Fact 2 *In the interval $[-1,1]$, $|x|$ can be approximated by $r_k(x)$ with $E_{R_{k+1}(x)}(|x|) \leq 3e^{-\sqrt{k}}$.*

This result is significant since no real polynomial of degree k can approximate $|x|$ with error $o(1/k)$. Also, this result is the starting point of rational approximation theory.

We use this result to show that the function computed by any threshold gate can be well approximated by a rational function of small degree. We say that a real multivariate function $g(x_1, x_2, \ldots, x_n)$ *approximates* a Boolean function $f(b_1, b_2 \ldots, b_n)$ with error E if $E = \max_{\vec{b}\in\{0,1\}^n} |f(\vec{b}) - g(\vec{b})|$.

Let G be a threshold gate on n inputs $\vec{w}$ be its weight vector. Assume that $f(\vec{b})$ is the Boolean function computed by G, that is, $f(\vec{b}) = \text{sgn}(\sum_{i=1}^n w_i b_i)$. Also, let $u = \sum_{i=1}^n |w_i|$. We assume $u \neq 0$. We then have the following approximation lemma for a single threshold gate.

Lemma 4 *$f(\vec{b})$ can be approximated by a real rational function $f(\vec{x})$ of degree $k+1$ with at most an error of $O(ue^{-\sqrt{k}})$.*

Proof: Consider the real multivariate function

$$h(\vec{x}) = (|\sum_{i=1}^n w_i x_i|/(\sum_{i=1}^n w_i x_i) + 1)/2.$$

Observe that h coincides with f when it is restricted to inputs from $\{0,1\}^n$. If h can be approximated by a rational function with an error E, then it would also be an approximation of f with an error E. Now consider $|(\sum_{i=1}^n w_i x_i)/u|$ whose value is in $[0,1]$ for $\vec{x} \in \{0,1\}^n$. By Newman's theorem, we can approximate this function by a real rational function $g'(\vec{x})$ of degree $k+1$ with at most an error of $3e^{-\sqrt{k}}$. Now, we use g' to get an approximation g for h.

$$g(\vec{x}) = (ug'(\vec{x})/(\sum_{i=1}^n w_i x_i) + 1)/2$$

It is clear that $g(\vec{x})$ is rational function of degree at most $k+1$. Notice that the absolute of $\sum_{i=1}^n w_i x_i$ is at least 1 for $\vec{x} \in \{0,1\}^n$ since all the w_i are integers and $\sum_{i=1}^n w_i x_i \neq 0$. Hence g approximates the Boolean function f with at most an error of $O(e^{-\sqrt{k}}/u)$.

We now use this lemma to show that a depth-2 threshold circuit can be computed by the sign of a real rational function whose degree is determined by the number of first level gates. Let T be a depth-2 threshold circuit computing a Boolean function $f(\vec{b})$. Let m is the number of non-constant inputs to the second level threshold gate in T. Also, for any threshold gate G in T, let u_G denote the sum of the absolute values of the weights of G. Let $u_{\max}$ denote the maximum value of u_G over all the gates G in T. We then have

Lemma 5 *$f(\vec{b}) = sgn(g(\vec{b}))$ where $g(\vec{x})$ is a real rational function in $x_1, x_2, \ldots, x_n$ of degree $m(k+1)$ as long as $u_{\max}^2 e^{-\sqrt{k}} < 1/3$.*

Proof: We approximate each non-constant input h_i of the second level gate by a degree $(k+1)$ multivariate rational function h_i' with at most an error of $3u_{max}e^{-\sqrt{k}}$. If $w_1, w_2, \ldots, w_m$ are the weights corresponding to the m non-constant inputs to the second-level gate of T, we define $g'(\vec{x})$ as

$$g'(\vec{x}) = \sum_{i=1}^{n} w_i h_i'(\vec{x}) + c.$$

where the constant c is contribution to the second-level gate of T from its constant inputs.

$g'(\vec{x})$ is a real rational function of degree at most $m(k+1)$ since it is a sum of m rational functions of degree $k+1$. Also, $g'(\vec{x})$ approximates $g(\vec{x}) = \sum_{i=1}^{n} w_i h_i(\vec{x}) + c$ with at most an error of $3u_{\max}^2 e^{-\sqrt{k}}$. Since $g(\vec{x})$ is an integer which is not equal to 0, the lemma follows.

Finally, we have the following theorem. Let T be any depth-2 threshold circuit that computes parity function on n inputs. Assume that all the weights in T are bounded by a polynomial in n.

Theorem 3 *The edge complexity of T is $\Omega(n^2/\log^2 n)$.*

Proof: We first prove an $\Omega(n/\log^2 n)$ lower bound on the node complexity of T. We then use lemma 1 to obtain our theorem.

Let m be number of first-level gates of T. We select $k = c\log^2 n$ for a sufficiently large c and apply lemma 5 to get a rational function $g(\vec{x}) = g_1(\vec{x})/g_2(\vec{x})$ of degree $O(m\log^2 n)$ such that $\text{sgn}(g(\vec{b}))$ is a parity function on $(b_1, b_2, \ldots, b_n)$. Observe that the condition in lemma 5 can be satisfied by selecting large enough c since all the weights of T are bounded by a polynomial.

Note that $\text{sgn}(g(\vec{b})) = \text{sgn}(g_1(\vec{b})g_2(\vec{b}))$ and g_1 and g_2 are multilinear polynomials when restricted to inputs 0 or 1. Hence, we have degree $O(m\log^2 n)$ multilinear polynomial which computes parity on n variables. Now, we use lemma 3 to conclude that $m = \Omega(n/\log^2 n)$.

References

[1] Alon, N., Bergmann, E.E., Coppersmith, D. and Odlyzko, A.M., Balancing Sets of Vectors, *IEEE Transactions on Information Theory*, **34**, 128–130, 1988.

[2] Baum, E.B., and Haussler, D., What Size Nets give Valid Generalization?, *Neural Computation*, **1**, 151–160, 1989.

[3] Beame, P., Brisson, E., and Ladner, R., The Complexity of Computing Symmetric Functions using Threshold Circuits, *Technical Report 90-01-01*, University of Washington, Seattle, 1990.

[4] Boppanna, R. and Sipser, M., Complexity of Finite Functions, Preliminary Draft, January 1988.

[5] Bruck, J., Harmonic Analysis of Polynomial Threshold Functions, Preprint, IBM Almaden Research Center, July 1988.

[6] Cottrell, G., Personal Communication, 1990.

[7] Hajnal, A. Maass, W., Pudlák, P., Szegedy, M. and Turán, G., Threshold

Circuits of Bounded Depth, *Proceedings of 28th Annual Symposium on Foundations of Computer Science*, 99–110, 1987.

[8] Hinton, G.E., Connectionist Learning Procedures, CMU-CS-87-115, Revised Version.

[9] Minsky, M. and Papert, S.A., *Perceptrons*, MIT Press, Cambridge, Massachusetts, 1969.

[10] Muroga, S., *Threshold Logic and its Applications*, New York-Wiley Interscience, 1971.

[11] Newman, D.J., Rational Approximation to $|x|$, *Michigan Math. Journal*, **11**, 11–14, 1964.

[12] O'Neil, P.E., Hyperplanes cuts of an n–cube, *Discrete Mathematics*, **1**, 193–195, 1971/72.

[13] Parberry, I. and Schnitger, G., Parallel Computation with Threshold Functions, *Journal of Computer and System Science* **36** 278–302, 1988.

[14] Reif, J., On Threshold Circuits and Polynomial Computations, *Proceedings of 2nd Structure in Complexity Theory Conference*, 118–125, 1987.

[15] Valiant, L.G., Functionality in Neural Nets, *First Workshop on Computational Learning Theory*, 24–33, 1988.

[16] Yao, A. C–C., Circuits and Local Computation, *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, 186–196, 1989.