Directed Monotone Contact Networks for Threshold Functions

Jaikumar Radhakrishnan K. V. Subrahmanyam

Computer Science Group
Tata Institute of Fundamental Research
Bombay, INDIA 400 005

Abstract

In this note we consider the problem of computing threshold functions using *directed* monotone contact networks. We give constructions of monotone contact networks of size $(k-1)(n-k+2)\lceil\log(n-k+2)\rceil$ computing T^n_k, for $2 \le k \le n-1$. Our upper bound is close to the $\Omega(kn\log(n/(k-1)))$ lower bound for small thresholds and the $k(n-k+1)$ lower bound for large thresholds. Our networks are described explicitly; we do not use probabilistic existence arguments.

Keywords. Computational complexity, contact networks, threshold functions.

1 Introduction and Definitions

A *monotone contact network* is a graph where each edge has a variable as its label. (In non-monotone networks, negated variables are also allowed to appear as labels.) For a pair (v, w) of vertices, the contact network computes the Boolean function $f_{(v,w)}$ as follows. On an assignment y, each edge is set to 0 or 1 according to the value of its label. Then $f_{(v,w)}(y) = 1$ if there is a path from v to w using only the edges with value 1, and $f_{(v,w)}(y) = 0$ otherwise. If N is a contact network with two distinguished vertices s (start) and t (terminal) then we refer to the function $f_{(s,t)}$ as the function computed by N. The *size* of a network is the number of edges in it.

Let n and k be positive integers such that $1 \le k \le n$. The k-th *threshold function* T^n_k is a Boolean function on n variables that takes the value 1 precisely when there are at least k 1's in the input. In this paper, we study the computation of threshold functions by *monotone contact networks.*

We now briefly describe the relationship between the monotone contact networks model and the more commonly studied models of monotone formulas and monotone circuits [BS]. Monotone contact networks are intermediate in power between monotone formulas and monotone circuits. Indeed, it is easy to see that every monotone formula can be converted to a monotone series-parallel contact network of the same size. However, monotone contact networks are much more powerful than monotone formulas, because they compute the (s,t)-connectivity function in linear size, while every monotone formulas computing this function has size $n^{\Omega(\log n)}$ [KW]. Yet, until recently, the best upperbounds for threshold functions in monotone contact networks models were often obtained from the corresponding results for monotone formulas. This, together with the fact that the computation of threshold functions in the monotone formulas model is not fully understood, provides the motivation for studying threshold function computation on monotone contact networks. The contact networks model is also related to the branching program model; we refer the reader to the papers of Razborov [Rz1, Rz2] for a description of this connection.

When the underlying graph of the contact network is undirected, such networks are called *undirected* monotone contact networks. Improving the $\Omega(n \log n / \log \log n)$ lower bound of Lupanov [L], Hansel [H] and Krichevskii [Kr] showed a lower bound of $\Omega(n \log n)$ on the size of such networks computing the threshold function T_2^n. Since monotone formulas can be converted to contact networks by representing the ANDs in series and the ORs in parallel, upper bounds for monotone formulas apply to undirected contact networks as well. Using the amplification method Boppana [B] showed that there exist formulas computing T_k^n of size $O(k^{4.3} n \log n)$. Applying the amplification method directly to monotone undirected contact networks, Dubiner and Zwick [DZ] constructed undirected monotone contact networks of size $O(n^{4.99})$, computing the majority function $T_{\lceil n/2 \rceil}^n$. Their method, when applied to other thresholds, yields $O(k^{3.99} n \log n)$ size undirected monotone contact networks computing T_k^n and T_{n-k+1}^n.

When the underlying graph is directed, such networks are called *directed* monotone contact networks. Since undirected networks can be converted to directed networks by replacing each undirected edge by a pair of directed edges, an $O(k^{3.99} n \log n)$ upper bound for computing T_k^n and T_{n-k+1}^n holds even in this model. By generalizing the lower bounds of Hansel and Krichevskii, Radhakrishnan [R] obtained a lower bound of $\lfloor k/2 \rfloor n \log(n/(k-1))$ on the size of any directed monotone network computing T_k^n, $2 \leq k \leq \frac{n}{2}$. If constant 1's are allowed to appear as labels in these networks, then they reduce to the monotone *contact-rectifier* networks considered by Markov [M]. The problem of computing threshold functions using monotone contact-rectifier networks was completely solved by Markov. He showed that the smallest such network computing T_k^n has size $k(n-k+1)$. Note that in the contact-rectifier networks model the edges with label 1 do not contribute to the size.

In this note we consider the problem of computing threshold functions using directed monotone contact networks without 1's. Since Markov's networks for computing T_1^n and T_n^n do not use 1's, we restrict our study to threshold functions T_k^n, for $2 \leq k \leq n-1$.

Our result. We eliminate the constant 1's in Markov's networks and obtain directed monotone contact networks of size $(k-1)(n-k+2)\lceil \log(n-k+2) \rceil$ computing T_k^n, for $2 \leq k \leq n-1$. For small thresholds, our upper bound is close to the the $\Omega(kn \log(n/(k-1)))$ lower bound; for large thresholds, it is close to the $k(n-k+1)$ lower bound (Markov's lower bound holds in this model). For computing *majority* this gives a network of size $O(n^2 \log n)$, whereas the best lower bound known is $\Omega(n^2)$. Our networks are described explicitly; the previously best upper bound known, obtained from the undirected networks of Dubiner and Zwick, used non-constructive arguments and gave directed networks of size $O(k^{3.99} n \log n)$.

2 Directed Networks for T_k^n

Markov's construction. Markov's network computing T_k^n can be described as an $(n-k+1) \times (k+1)$ grid. The top left corner of the grid is vertex s and the bottom right corner vertex t. Edges are directed from left to right along rows and from top to bottom along columns. The rows are numbered 1 to $n-k+1$ from top to bottom. The edges in row i have labels $x_i, x_{i+1}, \ldots, x_{i+k-1}$ from left to right. Every vertical edge has 1 as its label. Note that we may collapse the first column into s and the last column into t without changing the behavior of the network.

Eliminating the 1's. The 1's in Markov's construction can be replaced by $(n-k+1)$ edges in parallel with labels $x_1, x_2, \ldots, x_{n-k+1}$ respectively. The network continues to compute T_k^n and is of size $O(k(n-k+1)^2)$. This construction, while close to the lower bound for very large thresholds, is much inferior to the $O(k^{3.99} n \log n)$ upper bound for small thresholds. We show how the 1's can be eliminated more efficiently.

2.1 The Construction

We shall need the following notation. Let N be a network with l start vertices, $s_1, s_2, \ldots, s_l$, and l end vertices, $t_1, t_2, \ldots, t_l$. We say that N realizes $\mathsf{Selector}(l)$ on variables $(a_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l)$ if for $i, j = 1, 2, \ldots, l$,

$$f_{(s_i,t_j)} = \begin{cases} 0 & \text{if } i > j; \\ a_i \wedge b_j & \text{if } i \leq j. \end{cases}$$

For example, the network of Figure 1 realizes $\mathsf{Selector}(l)$ using 1's. Indeed, we may obtain

Figure 1: A directed network with 1's realizing $\mathsf{Selector}(l)$

a network N computing T_k^n, similar to Markov's, by composing $k-1$ networks, each realizing $\mathsf{Selector}(n-k+1)$. In the composition, the ith network N_i, $i = 1, 2, \ldots, k-1$, realizes $\mathsf{Selector}(n-k+1)$ on variables $(x_i, x_{i+1}, \ldots, x_{n-k+i}, x_{i+1}, x_{i+2}, \ldots, x_{n-k+i+1})$. To obtain the threshold network, collapse the vertices $s_1(N_1), s_2(N_1), \ldots, s_{n-k+1}(N_1)$ into one and call the resulting vertex $s(N)$, identify corresponding vertices of adjacent networks, that is, identify $t_i(N_j)$ with $s_i(N_{j+1})$ for $i = 1, 2, \ldots, n-k+1$ and $j = 1, 2, \ldots, k-2$, and collapse the vertices $t_1(N_{k-1}), t_2(N_{k-1}), \ldots, t_{n-k+1}(N_{k-1})$ into one vertex and call the resulting vertex $t(N)$. Thus, to obtain small networks computing T_k^n, it is sufficient to obtain small networks that realize $\mathsf{Selector}(n)$.

Let M be a network with l distinguished vertices $u_1, u_2, \ldots, u_l$. We say that M realizes $\mathsf{HalfSelector}(l)$ on variables $(a_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l)$ if it satisfies the following two conditions.

(i) For $i > j$, $f_{(u_i,u_j)} = 0$, that is, for $i > j$, there is no path from u_i to u_j.

(ii) For $i \leq j$, $a_i \wedge b_j \Rightarrow f_{(u_i,u_j)}$. That is, for $i \leq j$, there is a path from u_i to u_j all of whose labels come from $\{a_i, b_j\}$.

A network realizing $\mathsf{Selector}(l)$ can be obtained from a network realizing $\mathsf{HalfSelector}(l)$ by adding $2l$ edges as shown in Figure 2. We can obtain $\mathsf{Selector}$ networks directly without recourse to $\mathsf{HalfSelector}$ networks. However, the structure of the $\mathsf{Selector}$ networks obtained from $\mathsf{HalfSelector}$ networks will enable us to compose them more efficiently when we construct networks for threshold functions. In the following, we adopt the convention that the *empty graph* realizes $\mathsf{HalfSelector}(0)$.

Figure 2: A $\mathsf{Selector}(l)$ network

Lemma 1 *For all $n \geq 0$, there exists a directed contact network that realizes* $\mathsf{HalfSelector}(n)$ *with size at most*

$$(n+1)\lceil \log(n+1) \rceil - 2n.$$

Proof. We use induction on n. For $n = 0$, the empty graph gives the required contact network. Now let $r \geq 1$ and assume that for all n less than r, $\mathsf{HalfSelector}(n)$ can be realized by a network of size $(n+1)\lceil \log(n+1) \rceil - 2n$. We shall show that $\mathsf{HalfSelector}(r)$ can be realized by a network of size $(r+1)\lceil \log(r+1) \rceil - 2r$.

Let $n_1 = \lceil r/2 \rceil - 1$ and $n_2 = \lfloor r/2 \rfloor$. By our assumption there exists a network N_1, of size $(n_1+1)\lceil \log(n_1+1) \rceil - 2n_1$, realizing $\mathsf{HalfSelector}(n_1)$ on variables $(a_1, a_2, \ldots .a_{n_1}, b_1, b_2, \ldots, b_{n_1})$ and a network N_2, of size $(n_2+1)\lceil \log(n_2+1) \rceil - 2n_2$, realizing $\mathsf{HalfSelector}(n_2)$ on variables $(a_{n_1+2}, a_{n_1+3}, \ldots, a_r, b_{n_1+2}, b_{n_1+3}, \ldots, b_r)$. Then, the following network N (Figure 3), realizes $\mathsf{HalfSelector}(r)$ on variables $(a_1, a_2, \ldots, a_r, b_1, b_2, \ldots, b_r)$.

Figure 3: The induction step

Further, $\mathrm{size}(N) \leq (n_1+1)\lceil \log(n_1+1) \rceil - 2n_1 + (n_2+1)\lceil \log(n_2+1) \rceil - 2n_2 + (r-1)$. It can be verified that this is at most $(r+1)\lceil \log(r+1) \rceil - 2r$. This completes the induction step. ∎

Corollary 2 *For all $n \geq 1$, there exists a directed contact network that realizes* $\mathsf{Selector}(n)$ *with size at most* $(n+1)\lceil \log(n+1) \rceil$. ∎

Theorem 3 *For all n and k, $2 \leq k \leq n-1$, there exist directed monotone contact networks of size* $(k-1)(n-k+2)\lceil \log(n-k+2) \rceil$ *computing* T_k^n.

Proof. As observed earlier, a network computing T_k^n may be obtained by composing $(k-1)$ networks that realize $\mathsf{Selector}(n-k+1)$. By Corollary 2, there exist networks of size $(n-k+2)\lceil \log(n-k+2) \rceil$ that realize $\mathsf{Selector}(n-k+1)$. Hence, there exist directed monotone networks computing T_k^n with size at most $(k-1)(n-k+2)\lceil \log(n-k+2) \rceil$. ∎

The networks constructed above can be made a little more efficient. Observe that the edge (u_i, t_i) of stage j is in series with the edge (s_i, u_i) of the stage $j+1$, for $i = 1, 2, \ldots, n-k+1$ and $j = 1, 2, \ldots, k-2$. Further, these edges have the same label, x_{i+j}. By collapsing such edges into one, we obtain networks of size $(k-1)(n-k+2)\lceil \log(n-k+2) \rceil - (k-2)(n-k+1)$. For example, when $k = n-2$, this gives networks of size $8(n-3) - 3(n-4) = 5n - 12$, computing T_{n-2}^n. Such a network is shown in Figure 4. With $k = 2$, this construction gives networks

Figure 4: A directed contact network computing T_{n-2}^n

of size $n\lceil \log n \rceil$ computing T_2^n. The Hansel-Krichevskii method gives a lower bound of $n \log n$ (see [R]) for monotone directed networks computing T_2^n. This, together with the upper bound in Corollary 2, gives the following bounds on the size of $\mathsf{Selector}$ networks.

$$(n+1)\log(n+1) \leq \text{size}(\mathsf{Selector}(n)) \leq (n+1)\lceil \log(n+1) \rceil .$$

It follows that the $\mathsf{Selector}$ networks we obtain are close to optimal.

Acknowledgment

We are grateful to Shiva Chaudhuri, Magnús Halldórsson, Rakesh Sinha and the referees for their comments on the earlier versions of this paper.

References

[B] R. B. Boppana. *Amplification of Probabilistic Boolean Formulas.* Advances in Computing Research, Vol. 5, 1989, pp. 27–45.

[BS] R. B. Boppana and M. Sipser. *The Complexity of Finite Functions.* Chapter 14, The Handbook of Theoretical Computer Science, (J. van Leeuwen, ed.), Elsevier Science Publishers B. V., 1990, pp. 759–804

[DZ] M. Dubiner and U. Zwick. *Amplification and Percolation.* In the proceedings of the 33rd IEEE FOCS, 1992.

[H] G. Hansel. *Nombre minimal de contacts de fermature nécessaires pour réaliser une fonction booléenne symétrique de n variables.* C. R. Acad. Sci. Paris 258 (1964), pp. 6037–6040.

[Kr] R. E. Krichevskii. *Complexity of contact circuits realizing a function of logical algebra.* Sov. Phys. Dokl. 8 (1964) pp. 770–772.

[KW] M. Karchmer and A. Wigderson. *Monotone circuits for connectivity require superlogarithmic depth.* SIAM Journal of Discrete Mathematics, 3 (1990), pp.255–265.

[L] O. B. Lupanov.*On comparing the complexity of realization of monotone contact networks containing only closing contacts and by arbitrary contact networks.* Sov. Phys. Dokl. 7 (1962) pp. 486–489.

[MS] E. F. Moore and C. E. Shannon. *Reliable circuits using less reliable relays.* J. Franklin Inst. 262, 1956, pp. 191-208 and pp. 281–297.

[M] A. A. Markov. *On minimal switching-and-rectifier networks for monotone symmetric functions (in Russian).* In Problems of Cybernetics, vol. 8, pp. 117–121, Nauka, 1962.

[R] J. Radhakrishnan. *Better Bounds for Threshold Formulas.* In the proceedings of the 32nd IEEE FOCS, 1991, pp. 314–323.

[Rz1] A. A. Razborov. *Lower Bounds for Deterministic and Nondeterministic Branching Programs.* In the proceedings of the 8th FCT, Lecture Notes in Computer Science, 529, 1991, Springer-Verlag, New York/Berlin, pp. 47–60.

[Rz2] A. A. Razborov. *Lower bounds of the complexity of symmetric Boolean functions of contact-rectifier circuits.* Math. Notes Acad. Sci. USSR, 48(6), 1990, pp. 1226-1234.