Open Problems Column
Edited by William Gasarch

1 This Issues Column!

This issue's Open Problem Column is by William Gasarch and is *Wanted: A Top Down Proof for Circuit Lower Bounds on* PARITY$_n$.

2 Request for Columns!

I invite any reader who has knowledge of some area to contact me and arrange to write a column about open problems in that area. That area can be (1) broad or narrow or anywhere inbetween, and (2) really important or really unimportant or anywhere inbetween.

Wanted: A Top Down Proof for Circuit Lower Bounds on PARITY$_n$

.

By William Gasarch

We discuss a well known open problem and a well known approach to it which we want to make better known. The open problem is to find a top down proof for circuit lower bounds for PARITY$_n$. The approach is to use communication complexity.

3 Notation

We will use the following notations throughout this paper.

Notation 3.1 As usual the term *circuit* means *family of circuits*, one for each n. All circuits will alternate AND and OR gates. There will be no NOT gates; however, both inputs and their negations are available. The *depth* of a circuit is the longest path from input to output. The *fan-in* of a circuit is the max number of inputs to a gate. The *size* of a circuits is the number of gates. A (d, f, s)-*circuit* is a circuit of depth d, fan-in f, and size s. Note that $f \leq s$ so *unbounded fanin* really means $f = s$. Note also that $s \leq f^d$. It is clear what it means for a circuit to *compute a function* $g : \{0,1\}^n \to \{0,1\}^k$.

Notation 3.2 As usual the term *protocol* means *family of protocols*, one for each n. All protocols alternate who sends a message. The *number of rounds* in a protocol is the number of times someone sends a message. The *bits-per-round* of a protocol is the max number of bits in a message. An (r, b)-*protocol* is a protocol with r rounds and b bits-per-round. A protocol *computes a relation* $R \subseteq \{0,1\}^n \times \{0,1\}^n \times I$ if whenever Alice gets x and Bob gets y: (1) if there exists some $i \in I$ such that $(x, y, i) \in R$ then at the end of the protocol they agree on one such i, (2) if there is no such i then the protocol can do anything.

Note 3.3 An (r, b)-protocol can be represented by a tree (which we call a protocol tree) where (1) at the Alice nodes there are instructions on what to send, given her input and the prior messages she got, and there are 2^b outputs, one for each possible message, and (2) the Bob nodes are similar. The depth of the tree is r.

In the defintions of circuit (protocol) we insist that the AND-OR gates (Alice and Bob) alternate. This will not affect our point; however, there are cases where the constants can be improved by allowing non-alternation.

4 PARITY$_n$

Def 4.1 PARITY$_n$ is the function that, on input $(x_1, \ldots, x_n) \in \{0,1\}^n$ outputs $\sum_{i=1}^n x_i \pmod 2$.

How big a circuit do you need to computer PARITY$_n$? The following are known:

- There is an $(O(\log n), 2, O(n))$ circuit for PARITY$_n$ (easy).
- There is a $(O(1), 2^{O(n)}, 2^{O(n)})$ circuit for PARITY$_n$ (easy).
- For all constants d there is a a $(d, O(n^{(d-1)/(d-2)}2^{n^{1/(d-1)}}), O(n^{(d-1)/(d-2)}2^{n^{1/(d-1)}}))$ circuit for PARITY$_n$ (Håstad [4, 5, 6].)

Furst, Saxe, Sipser [3] (henceforth FSS) and Ajtai [1] showed that, for all constants d, for all polynomials $s(n)$, PARITY$_n$ cannot be computed by a $(d, s(n), s(n))$ circuit. (The motivation of FSS was to construct an oracle to separate PH from PSPACE; however, we do not consider that here.) The proof went as follows: For $d = 2$ this is easy to show. Assume there is a $(d, s(n), s(n))$ circuit where $s(n)$ is polynomial. Place a particular random restriction on the inputs where some inputs are set to 0, some are set to 1, and n' are left alone. With high probability the first two levels, say and AND of OR's, can be rewritten with only polynomial more gates as an OR of AND's. This collapses the two level of the circuit to one level. The result is a $(d-1, s'(n'), s'(n'))$ circuit for PARITY$_{n'}$. This is impossible inductively. A Lemma that says a random restriction leads to being able to write an AND or ORs as and OR of ANDS (without too much increase in size) is called *a switching lemma*.

Yao [13] proved that, for all d, PARITY$_n$ cannot be computed by a $(d, O(2^{n^{1/4d}}), O(2^{n^{1/4d}}))$ circuit. Håstad [4, 5, 6] obtained the optimal (up to polynomial factors) result: PARITY$_n$ cannot be computed by a $(d, O(2^{c_d n^{1/(d-1)}}), O(2^{c_d n^{1/(d-1)}}))$ circuit where $c_d = (0.1)^{d/(d-1)}$. Håstad obtained this result by refining the FSS-Ajtai's *switching lemma*. His version has been widely used in many contexts and, when *the switching lemma* is referred to, it means Håstad's. Razborov [11] obtained a different proof of the switching lemma that used a simpler logic. Fortnow and Laplante [2] recast Razborov's proof in terms of Kolmogorov Complexity.

The proofs of FSS, Ajtai, Yao, and Håstad are all *probabilistic*. Smolensky [12] had a completely different proof. He showed that PARITY$_n$ could not be approximated by a low degree polynomial, and that anything computed by a constant depth, small circuit could be. This leads to a lower bound on the size of d-depth circuits for PARITY$_n$ of $2^{\Omega(n^{1/2d})}$. This type of proof is called *algebraic*. Smolensky's proof also shows that (1) for all constants d, for all primes p, if you allow MOD-p gates then its still the case that any (d, s, s) circuits for PARITY$_n$ has exponential s, (2) similar for

computing MOD-q with AND, OR, NOT, and MOD-p gates where p is prime and q is relatively primes to p. All of these proofs are *bottom up* in that they begin at the input level.

Consider the case of $d = 3$. By Håstad's results:

- There is a $(3, O(n^2 2^{\sqrt{n}}), O(n^2 2^{\sqrt{n}}))$ circuit for PARITY_n.
- There is no $(3, O(2^{0.0316...\sqrt{n}}), O(2^{0.0316...\sqrt{n}}))$ circuit for PARITY_n. (The constant is really $(0.1)^{3/2}$.)

The lower bound was improved by Håstad, Jukna, Pudlak [7]. They showed that there is no $(3, O(2^{0.618...\sqrt{n}}), O(2^{0.618...\sqrt{n}}))$ circuit for PARITY_n (the constant is really $1/(\sqrt{2e}\ln 2)$). This proof was *top down* in that it began at the top gate. It is plausible that a top down proof of the lower bound for unbounded fan-in constant depth circuits for PARITY_n may lead to improved lower bounds. In the next section we discuss a top down approach via communication complexity.

5 Lower Bounds on Circuits via Communication Complexity

Karchmer [8] (see also Karchmer-Wigderson [9]) found a link between the depth of a circuit for a function g and the communication complexity of relation involving g. We present his theorem that links the two. Our treatment is from the excellent book on Communication Complexity by Kushilevitz and Nisan [10].

Def 5.1 Let $g : \{0,1\}^n \to \{0,1\}$. Let $R_g \subseteq \{0,1\}^n \times \{0,1\}^n \times [n]$ be the set

$$\{(x, y, i) : g(x) = 1 \wedge g(y) = 0 \wedge x_i \neq y_i\}.$$

We need the following well known lemma.

Lemma 5.2 *Let T be a $(d,1)$-protocol in the form of a binary protocol tree. Let w be a node of that tree. Let w_1 and w_2 be children of w.*

1. *There exists $A_w, B_w \subseteq \{0,1\}^n$ such that the set of inputs (x,y) that end up using node w is of the form $A_w \times B_w$.*
2. *Assume that w is a node where Alice sends a bit. Then there exists a partition $A_w = A_{w_1} \cup A_{w_2}$ and $B_w = B_{w_1} = B_{w_2}$.*
3. *Assume that w is a node where Bob sends a bit. Then there exists a partition $B_w = B_{w_1} \cup B_{w_2}$ and $A_{w_1} = A_{w_2} = A$.*

Theorem 5.3 *Let $g : \{0,1\}^n \to \{0,1\}$. There is a $(d, 2, 2^d)$ circuit for g iff there is a $(d,1)$ protocol for R_g.*

Proof:
1) Assume there is a $(d, 2, s)$ circuit for g. We use this to create a $(d,1)$-protocol for $D(R_g)$.

1. Alice takes x and runs the circuit on it. We assume she gets 1. Bob takes y and runs the circuit on it. We assume he gets 0. Note that Alice and Bob "disagree" on what the final output is.

2. We assume the top gate is an OR (if its an AND the proof is similar). Alice sees that the output when the circuit run on x is 1. Hence some input to that gate is 1. Alice sends Bob 0 if the left input is 1, and 1 if the right input is 1. Note that that input is the output of an AND gate. Note that Alice and Bob "disagree" on what the output of that AND gate is.

3. Bob sees that the output to the AND gate is 0. Hence some input to that gate is 0. Bob sends Alice 0 if the left input is 0, and 1 if the right input is 0. Note that that input is the output of an OR gate. Note that Alice and Bob "disagree" on what the output of that OR gate is.

4. They keep going in this matter until they find the original input they disagree on.

It is easy to see that this is a $(d, 1)$-protocol.

2) Assume there is a $(d, 1)$ protocol for R_g. Show there is a $(d, 2, s)$ circuit for g.

Take the protocol for R_g. Formally it is a binary protocol tree (since its a $(d, 1)$ protocol). We'll assume Alice sends first.

Take the tree and do the following to form a circuit:

1. Look at a node representing Alice sending a bit. In the protocol the node is thought of as *Alice knows her input and knows what Bob has send her and based on that sends a 0 or a 1.* As such it is a node with two edges coming out of it (going to Bob-nodes). We reverse that: we make this node into an AND gate and the edges are now the inputs.

2. Similarly, replace every Bob-node with an OR gate.

3. Let L be a leaf of the protocol. Associated to L is $i \in [n]$ such that, if the protocol gets to that leaf then Alice and Bob agree that $x_i \neq y_i$. Since the set of ordered pairs that goes to L is of the form $A_L \times B_L$ either: (1) every ordered pair that goes to L has $x_i = 1$ and $y_i = 0$, or (2) every ordered pair that goes to L has $x_i = 0$ and $y_i = 1$. If (1) replace L by input z_i, if (2) replace L by input $\overline{z_i}$.

We view every node in two ways: as a node in the protocol tree and as a gate in the circuit. For every node w let g_w be the function the circuit computes if it stopped at w. For every note w let A_w, B_w be as in Lemma 5.2.

By the definition of how we replace leaves with inputs, for all leaves L,

$$z \in A_L \implies g_L(z) = 1$$

$$z \in B_L \implies g_L(z) = 0$$

By induction, using Lemma 5.2, one can show that, for every node w

$$z \in A_w \implies g_w(z) = 1$$

$$z \in B_w \implies g_w(z) = 0$$

Note that for the top gate w, $A_w = g^{-1}(1)$ and $B_w - g^{-1}(0)$. Hence the circuit computes g. ∎

Before this result the following thoughts had currency:

- We've made very little progress in proving lower bounds on circuits.
- We've made a lot of progress in proving lower bounds on communication protocols.

So the hope was that Theorem 5.3 would allow progress on lower bounds on circuits. Did it? Yes and mostly No. There were many results about *monotone circuits* that used this framework. But alas, very few on general circuits.

A proof of a circuit lower bounds that used Theorem 5.3 would almost surely be a top down proof.

6 What about Known Circuit Results?

Could there be a communication complexity proof that any constant depth, unbounded fan-in circuit for PARITY_n requires exponential size? Theorem 5.3 is about fan-in 2 circuits. We need a version for unbounded fan-in circuits.

Theorem 6.1 *Let $g : \{0,1\}^n \to \{0,1\}$. There is a (d, s, s) circuit for g iff there is a $(d, \lceil \lg(s) \rceil)$ protocol for R_g.*

Proof:

All $\lg(s)$ in this proof are really $\lceil \lg(s) \rceil$.

1) Assume there is a (d, s, s) circuit g. The $(d, \lg(s))$ protocol for R_g is similar to the $(d, 1)$ protocol in the proof of Theorem 5.3.1. except that, for Alice (Bob) to specify which input was 1 (0) takes $\lg(s)$ bits. It is easy to see that this is a $(d, \lg(s))$-protocol.

2) Assume there is a $(d, \lg(s))$ protocol for R_g. The (d, s, s) circuit for g is similar to the circuit in the proof of Theorem 5.3.2. ∎

7 Open Problem

Theorem 6.1 points to a possible alternative method to get lower bounds on the size of constant depth circuits for PARITY_n:

Conjecture 7.1 *Fix d, a constant. If there is a (d, b) protocol for R_{PARITY_n} then b is $\Omega(2^{n^{1/(d-1)}})$.*

There is one thing wrong with this conjecture. It is already known to be true! Just use Theorem 6.1.2 and use the known lower bounds on PARITY_n. So what are we really looking for? **Open Problem:** Give a communication complexity proof for any of the following. Fix d, a constant.

1. If there is a (d, b) protocol for R_{PARITY_n} then b is superpolynomial.
2. If there is a (d, b) protocol for R_{PARITY_n} then b is $\Omega(2^{n^{1/(d-1)}})$ (or replace $2^{n^{1/(d-1)}}$ with a smaller superpolynomial function).

Such a proof would give another way to obtain lower bounds on PARITY_n. While this is a good end in itself, we note that it might also lead to better lower bounds.

8 Acknowledgments

We would like to thank Paul Beame, Lance Fortnow, Eyal Kushilevitz, and Emanuele Viola for helpful email discussions.

References

[1] M. Ajtai. Σ_1^1-formulae on finite structures. *Annals of Pure and Applied Logic*, 24:1–48, 1983.

[2] L. Fortnow and S. Laplante. Circuit lower bounds a la Klomogrove. *Information and Computation*, 15, 1995.

[3] M. Furst, J. B. Saxe, and M. Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, April 1984.

[4] J. Håstad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the Eighteenth Annual ACM Symposium on the Theory of Computing,* Berkeley CA, pages 6–20, 1986.

[5] J. Håstad. *Computational limitations of small-Depth Circuits.* MIT Press, Cambridge, MA, 1987.

[6] J. Håstad. Almost optimal lower bounds for small depth circuits. In S. Micali, editor, *Randomness and Computation*, pages 143–170, Greenwich, CT, 1989. JAI Press.

[7] J. Håstad, S. Jukna, and P. Pudlak. Top down lower bounds for depth-three circuits. *Computational Complexity*, 5, 1995.

[8] M. Karchmer. *Communication complexity: A new approach to circuit depth.* MIT Press, Cambridge, MA, 1989.

[9] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require superlogarthmic depth. *SIAM Journal on Discrete Mathematics*, 3(2):255–265, 1990. Earlier version in STOC 1988.

[10] E. Kushilevitz and N. Nisan. *Communication Complexity.* Cambridge University Press, Cambridge, England, 1997.

[11] A. Razborov. Bounded arithmetic and lower bounds in Boolean complexity. In P. Clote and J. Remmel, editors, *Feasible Mathematics II*, pages 344–386, 1995.

[12] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proceedings of the Nineteenth Annual ACM Symposium on the Theory of Computing,* New York, pages 77–82, 1987.

[13] A. C. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings of the 26th Annual IEEE Symposium on Foundations of Computer Science,* Portland OR, pages 1–10, 1985.