On the Bottleneck Counting Argument*

Janos Simon
Dept of Computer Science
University of Chicago
Chicago, IL 60637, USA
simon@cs.uchicago.edu

Shi-Chun Tsai
Information Management Department
National Chi-Nan University
Pu-Li, Nan-Tou 545, TAIWAN
tsai@csie.ncnu.edu.tw

January 26, 1999

Abstract

Both the bottleneck counting argument [7, 8] and Razborov's approximation method [1, 4, 12] have been used to prove exponential lower bounds for monotone circuits. We show that under the *monotone circuit* model for every proof by the approximation method, there is a bottleneck counting proof and vice versa. We also illustrate the elegance of the bottleneck counting technique with a simple self-explained example: the proof of a (previously known) lower bound for the 3-CLIQUE$_n$ problem by the bottleneck counting argument.

Keywords: Computational complexity; Circuit complexity; Monotone circuit complexity

1 Introduction

Razborov's proof of an exponential lower bound on the size of monotone Boolean circuits to detect cliques in a graph [12] [1], represented a breakthrough in the theory of monotone circuit complexity. The proof introduced the method of approximation. The method is roughly as follows. Consider two sets of test inputs, a positive (the output is 1) and a negative one. Given a monotone circuit $\mathcal{M}$ that computes a monotone function, one replaces each gate E in $\mathcal{M}$ with an approximator $\tilde{E}$ level by level from the bottom level to the top. This produces a circuit $\tilde{\mathcal{M}}$ that computes a function that approximates the function that $\mathcal{M}$ computes. To obtain the lower bound on the size of $\mathcal{M}$ one needs to show that for each E, $\tilde{E}$ introduces only a small fraction of extra errors, and $\mathcal{M}$ disagrees with $\tilde{\mathcal{M}}$ on a large fraction of the test inputs. Since each gate introduces only a small number of new errors and the number of errors for the whole circuit is large, $\mathcal{M}$ must have many gates. Many more lower bounds were proven using the approximation method, for example Yao [15], Goldmann and Håstad [6].

The bottleneck counting argument, introduced by Haken [7], defines a mapping from a subset of the inputs to the gates in the circuit. The number of inputs that are mapped to the gates divided by the maximum number of inputs that can be mapped to a gate in the circuit is the lower bound of the circuit size. Thus by finding a proper mapping, we can show some non-trivial lower bounds.

*A preliminary version of this paper appeared in *Proceedings of the 12th Annual IEEE Conference on Computational Complexity*, Ulm, Germany, 1997, pp. 297–301

Recently Haken [8] has applied this argument to prove an exponential lower bound on the size of monotone circuits for the Broken Mosquito Screen problem, which is a special version of the CLIQUE problem. Similar lower bounds have been proven earlier by Razborov [12] and later strengthened by Andreev [4], and Alon and Boppana [1] using the method of approximation. While the result is old, the bottleneck method provides a new simple proof for this strong lower bound. Chronologically, the two methods appeared in publications about the same time, 1985. Since then the approximation method has drawn much more attention and it has evolved into a standard method for proving lower bounds for monotone circuits.

In this note we show the equivalence of these two methods. I.e., we show that for any lower bound proved by the approximation method there is a corresponding bottleneck counting proof and vice versa. The consequence is that the bottleneck counting method does not really help to prove what is beyond the reach of the approximation method. Razborov [13], shows strong limits to the applicability of the approximation method. The bottleneck counting method has the same limitations.

We also show a lower bound for the 3-CLIQUE$_n$ problem to illustrate the bottleneck counting method. The 3-CLIQUE$_n$ is the problem of determining if an undirected graph G of n vertices has a triangle or not, which is simply a special case of the k-CLIQUE$_n$ problem, whose exponential monotone lower bound has been proven in [8, 12]. We consider the complexity of computing 3-CLIQUE$_n$ using monotone circuits with $\vee$-gates and $\wedge$-gates of fan-in 2. A graph of n vertices can be encoded by $\binom{n}{2}$ variables $x_{i,j}$, which is 1 if there is an edge between vertices i and j, and is 0 otherwise. It is clear that 3-CLIQUE$_n$ has an upper bound $\binom{n}{3}$. The lower bound $(\Omega(\frac{n^3}{\log^4 n}))$ for this problem has been proven in [2] by Razborov's approximation method [1, 4, 12]. Using the bottleneck counting argument [7, 8], we give a new and simpler proof that achieves the same lower bound. We believe that there are two good reasons to present the proof. First, it is considerably simpler than the one in [1, 4, 12]. Second, it shows that Haken's argument can be used to prove rather tight polynomial lower bounds. Previous uses of the bottleneck argument prove exponential lower bounds. Karchmer [10] also obtained exactly the same bound for 3-CLIQUE$_n$ by the fusion method. This prompts an open question that if the fusion method is equivalent to the above two methods.

The result of this paper has also been observed independently by Amano and Maruoka [3], Berg and Ulfberg [5] and Jukna [9]. However, this paper is the first one to formalize this observation.

2 Approximation and bottleneck counting methods

A Boolean function $f(x_1, ..., x_n)$ is monotone if $f(x_1, ..., x_n) = 1$ and $x_i \leq y_i$ for all $1 \leq i \leq n$ implies $f(y_1, ..., y_n) = 1$. A binary monotone circuit uses only AND-gates and OR-gates. We will measure the difficulty of a monotone Boolean function by the size (number of gates) of a monotone circuit that computes it.

Let f be an n-variable monotone boolean function. Let G_0 (B_0) be the set of test inputs that make f output 1 (0). In addition, we require that each element in G_0 (B_0) be a minterm (maxterm). Recall that (for monotone functions f) a set S of variables is a *minterm* if setting all $x \in S$ to 1 forces f to have the value 1, but no proper subset of S has this property. It suffices to prove the lower bound for circuits that separate G_0 and B_0. Let $\mathcal{M}$ be a monotone circuit that computes f. Without loss of generality we can assume $\mathcal{M}$ is a leveled circuit and that the output is on the top level.

2.1 Approximation method: schematic outline.

To apply the approximation method, one replaces each gate E in $\mathcal{M}$ by an approximator $\tilde{E}$ from the bottom level to the top inductively. The approximator $\tilde{\mathcal{M}}$ of the output gate will output an approximation for the circuit $\mathcal{M}$. To prove a lower bound we need to show that for each E, $\tilde{E}$ introduces errors only on a small fraction of test inputs, while $\mathcal{M}$ and $\tilde{\mathcal{M}}$ disagree on a large fraction of the test inputs. If there are such approximators, then the size of $\mathcal{M}$ must be large.

Formally [1, 2, 12, 13, 14], for any Boolean function $f(x_1, ..., x_n)$, define $A(f) = \{(x_1, ..., x_n) \in \{0, 1\}^n : f(x_1, ..., x_n) = 1\}$. It suffices to compare two Boolean functions by looking at their pre-images of 1. It is clear that $A(f \vee f') = A(f) \cup A(f')$, and $A(f \wedge f') = A(f) \cap A(f')$. A monotone circuit $\mathcal{M}$ of size s that computes f corresponds to a straight-line program of length s computing f, i.e., a sequence of functions $x_1, ..., x_n, E_1, ..., E_s$, where $E_s = f$ and each E_i, for $1 \leq i \leq s$, takes two previous functions of the sequence as inputs. Note that we will use E_i to denote both a gate and the Boolean function computed by the subcircuit of $\mathcal{M}$ rooted at E_i. Thus we obtain a sequence of subsets of $\{0, 1\}^n : A_{-n} = A(x_n), ..., A_{-1} = A(x_1), A_1, ..., A_s$, where $A_i = A(E_i)$, for $1 \leq i \leq s$ and for each A_i, it is either the union or the intersection of previous two subsets. We replace the sets A_i by approximated sets M_i as follows. Let $M_{-i} = M_{x_i} = A(x_i)$ for $1 \leq i \leq n$ and let M_i approximate A_i by replacing the union and intersection with approximators $\sqcup$ and $\sqcap$, respectively. For any two sets $M, L \subseteq \{0, 1\}^n$, we require that

$$M \sqcup L \supseteq M \cup L \quad \text{and} \quad M \sqcap L \subseteq M \cap L. \tag{1}$$

Suppose $j > k, \ell$. If $A_j = A_\ell \cup A_k$, then define $M_j = M_\ell \sqcup M_k$. If $A_j = A_\ell \cap A_k$, then $M_j = M_\ell \sqcap M_k$. For a successful approximation proof we need that both $\delta^j_\sqcup := M_j - (M_\ell \cup M_k)$ and $\delta^j_\sqcap := (M_\ell \cap M_k) - M_j$ be *small* for all E_j's in the straight-line program, and $M_s \bigtriangleup A(f)$ be *large*, where $\bigtriangleup$ is the operator of symmetric difference. The precise definition of the approximators varies depending on the specific problems. The following lemma is crucial for the approximation method.

Lemma 2.1 *[1, 2, 12, 13, 14] For all M_i's, $A(E_i) - (\cup_{j \leq i} \delta^j_\sqcap) \subseteq M_i \subseteq A(E_i) \cup (\cup_{j \leq i} \delta^j_\sqcup)$.*

Lemma 2.1 holds for any choice of the approximate operations $\sqcap$ and $\sqcup$ that satisfy relation (1). Now, if we can show that δ's have an upper bound ϵ and that $L = |A(f) \bigtriangleup M_s|$ is *big*, then $s \geq L/\epsilon$ which must be *large*.

2.2 Bottleneck counting method: schematic outline.

The key idea of the bottleneck counting method is to define a mapping μ from a subset S of $G_0 \cup B_0$ to the gates of $\mathcal{M}$ such that for any gate E in $\mathcal{M}$, the ratio of $|S|/|\mu^{-1}(E)|$ is large. The mapping μ is defined sequentially. First one element of $G_0 \cup B_0$ is mapped and then that element is deleted from $G_0 \cup B_0$ yielding the set $G_1 \cup B_1$. The procedure continues level by level from the bottom to the top until the set $G_j \cup B_j$ becomes too small for the further specification of μ to make sense. The set S is then $(G_0 \cup B_0) \setminus (G_j \cup B_j)$. To prove a lower bound, it will be shown that "not too many" elements of S can be mapped to any one gate in $\mathcal{M}$.

To define the mapping μ, Haken uses a measure of "progress". An input vector g is mapped to a gate E (which is the root of a subcircuit of $\mathcal{M}$) at which the circuit $\mathcal{M}$ makes particular progress in classifying g. Further arguments show that progress is not made for too many graphs at that same gate E. For some problems, the desired measure of progress turns out to be "the length of a minimal fence" [8]. We define the fences as certain problem-dependent progressive properties. The measure of progress depends on the fences, and it can be any meaningful measurement that defines a good mapping in the sense of getting a nontrivial lower bound.

Definition 2.2 *Let E be a gate in $\mathcal{M}$ and let g be an input in the set G_i. A fence around g at gate E and at time i is a property $\mathcal{P}(E, g, i)$. Dually, for a gate E in $\mathcal{M}$ and b in B_i, a fence around b at gate E and at time i is a property $\mathcal{Q}(E, b, i)$.*

For example, let $F_{E,g,i}$'s and $F_{E,b,i}$'s be conjunctive and disjunctive Boolean formulas, respectively. In Haken's paper [8], at time i, for a gate E and an input $g \in G_i$, $\mathcal{P}(E, g, i)$ is defined as: $E(g) = 1$ and $F_{E,g,i}(g) = 1$ and $((\forall b \in B_i)[(E(b) = 0) \Rightarrow (F_{E,g,i}(b) = 0)])$ and all such $F_{E,g,i}$'s have at least $n/2$ literals. In other words, $F_{E,g,i}$ is required to compute just as good a separation of g from the set B_i, as does the gate E. Similarly, for an input $b \in B_i$, $\mathcal{Q}(E, b, i)$ is defined as: $E(b) = 0$ and $F_{E,b,i}(b) = 0$ and $((\forall g \in G_i)[(E(g) = 1) \Rightarrow (F_{E,b,i}(g) = 1)])$ and all such $F_{E,b,i}$'s have at least $n/2$ literals. So $F_{E,b,i}$ does just as good a job as does E on b and on G_i.

Next we outline the procedures to construct a mapping via properties $\mathcal{P}$'s and $\mathcal{Q}$'s.

Construction:
Let E_0 be the gate at the lowest level, and leftmost within the level in $\mathcal{M}$, such that there exists an input $d_0 \in G_0 \cup B_0$ that satisfies $\mathcal{P}(E_0, d_0, 0)$ or $\mathcal{Q}(E_0, d_0, 0)$. Here the specification of leftmost is not important, any consistent ordering will do. Define $\mu(d_0)$ to be E_0 and let $(G_1 \cup B_1)$ be $(G_0 \cup B_0)$ with d_0 taken out. Now repeat the process while possible: at time i let E_i be the lowest level and leftmost gate in $\mathcal{M}$ such that there is an input $d_i \in (G_i \cup B_i)$ satisfying $\mathcal{P}(E_i, d_i, i)$ or $\mathcal{Q}(E_i, d_i, i)$. Define $\mu(d_i)$ to be E_i and delete d_i from $(G_i \cup B_i)$ to get $(G_{i+1} \cup B_{i+1})$. Eventually, the remaining inputs do not satisfy either property, and no more are mapped.

From the above we notice that it is crucial to find proper test input sets and the required progressive properties for the bottleneck counting approach to yield a good lower bounds. These are problem-dependent tasks. In the approximation method we had the problem-dependent difficulty of finding good approximators. We prove that these are equally difficult tasks: given a bottleneck proof we can produce an equivalent approximation method proof and vice versa.

2.3 Proof of equivalence

Theorem 2.3 *A proof of lower bound for a monotone circuit by the approximation method yields a proof of the same bound by the bottleneck counting method, and vice versa.*

Proof. $(\Rightarrow)$: From the approximation method we know that there is an approximator M_i for each $A(E_i)$, where E_i is a gate in $\mathcal{M}$ and $\frac{|A(f) \triangle M_s|}{\max_j(|\delta^j_\sqcap|, |\delta^j_\sqcup|)}$ gives a lower bound for the circuit size s. We construct a mapping for the bottleneck counting proof from the approximation proof. Define the progressive property $\mathcal{P}(E_j, g, i)$ to be $(g \in (\delta^j_\sqcap \cap G_i))$, if E_j is an AND-gate; otherwise $(g \in (\delta^j_\sqcup \cap G_i))$. Similarly, the progressive property $\mathcal{Q}(E_j, b, i)$ is defined as $(b \in (\delta^j_\sqcap \cap B_i))$, if E_j is an AND-gate; otherwise $(b \in (\delta^j_\sqcup \cap B_i))$. Note that if $d \in \delta^j_\sqcap - (G_i \cup B_i)$, then d has been mapped earlier, similarly for $d \in \delta^j_{\sqcup} - (G_i \cup B_i)$.

Let E_0 be the gate at the lowest level, and leftmost within the level in $\mathcal{M}$. Let D_0 be the set of inputs in $G_0 \cup B_0$ such that all $d \in D_0$ satisfy $\mathcal{P}(E_0, d, i)$ or $\mathcal{Q}(E_0, d, i)$ for $i = 0$. Then we define, for all $d \in D_0$, $\mu(d)$ to be E_0 and let $G_1 := G_0 - D_0$ and $B_1 := B_0 - D_0$. After this, we say E_0 is mapped, i.e., no more input will be mapped to this gate. Similarly, let E_i be the lowest level and leftmost unmapped gate in $\mathcal{M}$ and D_i be the set of inputs in $G_i \cup B_i$ such that $\mathcal{P}$ or $\mathcal{Q}$ is satisfied. Define $\mu(d)$ to be E_i for all test input d in D_i and update $G_i \cup B_i$. We repeat the above until there is no more input satisfying the properties at any gates and obtain a mapping μ from the inputs to the gates.

By the assumption, there are $|M_s \triangle A(f)|$ inputs mapped, and at each gate there are at most $\max_j(|\delta^j_{\sqcap}|, |\delta^j_{\sqcup}|)$ inputs mapped. Therefore the induced bottleneck counting proof yields the same lower bound as the one obtained by the approximation method.

($\Leftarrow$) : Now suppose we have a bottleneck counting proof for a monotone circuit lower bound. Let $\mathcal{P}$'s and $\mathcal{Q}$'s be the properties used for constructing the mapping μ in the proof. We want to construct the approximation sets of $A(E_i)$'s. It is clear that $M_{x_i} = A(x_i)$. By induction, suppose we have M_j's for smaller j's. Suppose $j > \ell, k$. Let $E_j := E_\ell \wedge E_k$ (i.e., E_j is an AND-gate) and let M_ℓ, M_k be the corresponding approximate sets. Let $\delta^j_{\sqcap} := \mu^{-1}(E_j)$. We define $M_j = M_\ell \sqcap M_k := (M_\ell \cap M_k) - \delta^j_{\sqcap}$. If E_j is an OR-gate, we define $\delta^j_{\sqcup} := \mu^{-1}(E_j)$ and $M_j = M_\ell \sqcup M_k := (M_\ell \cup M_k) \cup \delta^j_{\sqcup}$. It is clear that the M_i's satisfy relation (1). Note that both $\delta^j_{\sqcap}$ and $\delta^j_{\sqcup}$ have 'small' cardinality. Since in the bottleneck counting proof many inputs are mapped and by Lemma 2.1 M_s deviates from $A(f)$ by a large fraction, we know $|A(f) \triangle M_s|$ is large. Hence the lower bound obtained by the approximation method is the same as the one proven by the bottleneck counting method. □

The elegance and comparative simplicity of Haken's paper yielded some optimistic speculation that perhaps one would be able to achieve new breakthroughs. The equivalence of the two methods, together with Razborov's negative results makes this unlikely.

3 Lower bound on the 3-CLIQUE problem

Given a graph $G(V, E)$ with $|V| = n$, we want to test if G has a triangle or not, using a monotone circuit. First let V be any n vertex set. Let G_0 be the set of all graphs over V with exactly one triangle. We call the graphs in G_0 'good graphs'. It is clear that $|G_0| = \binom{n}{3}$. Let B_0 be the set of complete bipartite graphs over $X \cup Y$, where $X \cup Y = V$, and X, Y are non-empty. It is clear that graphs in B_0 do not have a triangle and $|B_0| = 2^{n-1} - 1$. We call the graphs in B_0 'bad graphs'. It is sufficient to prove the lower bound for the monotone circuit that separates G_0 and B_0.

Let $\mathcal{M}$ be a monotone circuit that separates G_0 and B_0. We will define a mapping μ, as in Haken's paper [8], from a subset of $G_0 \cup B_0$ to the gates of $\mathcal{M}$. For any gate E in $\mathcal{M}$ and any input instance g, we define $E(g)$ as the output of the subcircuit rooted at E with input g.

Definition 3.1 *[8] Let $E \in \mathcal{M}$, $g \in G_i$, $E(g) = 1$, and y_i's be positive literals. A fence of g and E at step i is a conjunctive formula C of the form $y_1 \wedge ... \wedge y_q$, such that $C(g) = 1$ and $\forall b \in B_i, E(b) = 0 \Rightarrow C(b) = 0$. A minimal fence C of g and E at time i is the shortest fence that satisfies the above definition. Similarly, a fence at time i for $b \in B_i, E(b) = 0$ and $E \in \mathcal{M}$*

is a disjunctive formula D of the form $y_1 \vee ... \vee y_r$ such that $D(b) = 0$ and $\forall g \in G_i, E(g) = 1 \Rightarrow D(g) = 1$.

Next we construct the map μ by the following procedure.

1. Initialization: $i := 0$.
2. For $g \in G_i \cup B_i$, define $\mu(g) := E$, if E is the lowest and leftmost gate in $\mathcal{M}$, such that 1) if E is an AND-gate, then $E(g) = 1$ and g has a minimal fence of size at least 2, or 2) if E is an OR-gate, then $E(g) = 0$ and g has a minimal fence of size at least $4.5 \log^2 n$.
3. Let $G_{i+1} \cup B_{i+1} := G_i \cup B_i - \{g\}$.
4. Let $i := i + 1$ and repeat step 2, 3, and 4 until there is no fence of the size required in step 2.

We prove the lower bound as follows.

Lemma 3.2 *At least $\binom{n}{3} - (4.5 \log^2 n)(n-2)$ graphs in G_0 or all of the graphs in B_0 are mapped by μ.*

Proof. There are two possible cases. If all of the graphs in G_0 or in B_0 are mapped, then there is nothing to prove and the claim is true.

If there are graphs in G_0 and B_0 that are not mapped, then consider the output gate E of $\mathcal{M}$ and an unmapped graph $b \in B_0$. By the mapping procedure, we know b has a short fence D at E, such that $E(b) = 0$ and $D(g) = 1$ for all unmapped $g \in G_0$. This means that each unmapped $g \in G_0$, has at least one edge corresponding to a variable in D. Note that the number of graphs in G_0 that contain a specific edge is $(n-2)$, since there are $n-2$ possible ways to choose the third vertex of a triangle. In total there are at most $(4.5 \log^2 n)(n-2)$ graphs in G_0 that contain at least one edge corresponding to a variable in D. Therefore, at least $\binom{n}{3} - (4.5 \log^2 n)(n-2)$ good graphs are mapped. This completes the proof. □

The following lemma is true for all monotone circuits that use the mapping procedure above.

Lemma 3.3 *[8] Each $g \in G_0$ can only be mapped to an $\wedge$-gate. Each $b \in B_0$ can only be mapped to an $\vee$-gate.*

Lemma 3.4 *At most $(4.5 \log^2 n)^2$ good graphs can be mapped to a single $\wedge$-gate.*

Proof. Let E be an $\wedge$-gate and g be any graph in G_0 with $\mu(g) = E$. Let $b_1, ..., b_s$ be all the bad graphs with $E(b_i) = 0$. Then each b_i has a short fence D_i (of size at most $(4.5 \log^2 n) - 1$) such that $D_i(g) = 1$. For each g satisfying $D_i(g) = 1$, it must have at least one edge corresponding to the variables in D_i for $i = 1..s$, and it has at least 2 different edges corresponding to the variables in $D_1, ..., D_s$, otherwise g will have a short fence. Since g has only three edges, g will be uniquely determined by two edges. Therefore there are at most $(4.5 \log^2 n)^2 = (4.5)^2 \log^4 n$ such graphs in G_0. So at most $(4.5 \log^2 n)^2$ good graphs can be mapped to a single $\wedge$-gate. □

Lemma 3.5 *At most $2^{n - \sqrt{2(4.5 \log^2 n)} - 1}$ bad graphs can be mapped to a single $\vee$-gate.*

Proof. Let E be an $\vee$-gate and b be any graph in B_0 with $\mu(b) = E$. Let $g_1, ..., g_r$ be all the good graphs with $E(g_i) = 1$. Then each g_i has a short fence C_i (of size 1) such that $C_i(b) = 0$. Among these r short fences there must be at least $(4.5 \log^2 n)$ different variables, since b must have a long minimal fence in order to be mapped on E. For each b satisfying $C_i(b) = 0$, it must have no edge corresponding to the variable in each C_i for $i = 1..r$. These $(4.5 \log^2 n)$ anti-edges contain at least $\sqrt{2(4.5 \log^2 n)}$ vertices. Since b is a complete bipartite graph, these $\sqrt{2(4.5 \log^2 n)}$ vertices must be fixed in either X or Y. There are at most $2^{n-\sqrt{2(4.5\log^2 n)}-1}$ such bipartite graphs. Therefore there are at most $2^{n-\sqrt{2(4.5\log^2 n)}-1}$ bad graphs that can be mapped to a single $\vee$-gate. □

Theorem 3.6 *The monotone complexity of 3-CLIQUE is* $\Omega(\frac{n^3}{\log^4 n})$.

Proof. By Lemma 3.2, 3.4, and 3.5, we know the size of $\mathcal{M}$ must be at least $\min\{\frac{\binom{n}{3} - 4.5\log^2 n(n-2)}{(4.5\log^2 n)^2}, \frac{2^{n-1}-1}{2^{n-\sqrt{2(4.5\log^2 n)}-1}}\}$. By simple calculations, the lower bound is $\Omega(\frac{n^3}{\log^4 n})$. This completes the proof. □

4 Conclusion and remarks

We proved that the bottleneck counting argument and the method of approximation imply each other. This shows that the inherent limit of the approximation method is inherited by the bottleneck counting method. Often the bottleneck counting method gives a better insight on the problem. The proof of the monotone lower bound for the 3-CLIQUE problem illustrates the technique of the bottleneck counting argument.

It would be interesting to get matching upper and lower bounds for the 3-CLIQUE problem, or other polynomial size monotone problems.

An important related open problem is the size of *noisy* [11] monotone circuits for such problems. The upper bound is $S \log S$ where S is the size of the non-noisy circuit. It is not known whether this bound is sharp for $S = \Omega(n)$.

Acknowledgments We would like to thank Eric Allender for reminding us the work of Karchmer and the anonymous referees for helpful comments.

References

[1] N. Alon and R. B. Boppana. The monotone circuit complexity of boolean functions. *Combinatorica*, 7(1): 1–22, 1987.

[2] N. Alon, J. Spencer, and P. Erdős. *The Probabilistic Method.* John Wiley and Sons Inc., 1991.

[3] K. Amano and A. Maruoka Potential of the Approximation Method In *Proc. 37th Symposium on Foundations of Computer Science*, pages 431–440, 1996.

[4] A. E. Andreev. On a method for obtaining lower bounds for the complexity of individual monotone functions. *Soviet Math. Dokl*, 31:530–534, 1985.

[5] C. Berg and S. Ulfberg. Symmetric approximation arguments for monotone lower bounds without sunflowers. to appear in *Computational Complexity.*

[6] M. Golmann and J. Håstad. Monotone circuits for connectivity have depth $(\log n)^{2-o(1)}$. In *Proc. 27th Annual ACM Symposium on the Theory of Computing*, pages 569–574, 1995.

[7] A. Haken. The intractability of resolution. *Theoret. Comput. Sci.*, 39:297–308, 1985.

[8] A. Haken. Counting bottlenecks to show monotone P≠NP. In *Proc. 36th Symposium of Foundations of Computer Science*, pages 36–40, 1995.

[9] S. Jukna. Finite limits and monotone computations over reals. In *Proceedings of the 12th Annual IEEE Conference on Computational Complexity*, pages 302–313, 1997.

[10] M. Karchmer. On proving lower bounds for circuit size. In *Proceedings of the 8th Annual IEEE Conference of Structure in Complexity Theory*, pages 112–118, 1993.

[11] N. Pippinger. On networks of noisy gates. In *Proc. 26th Symposium on Foundations of Computer Science*, pages 30–38, 1985.

[12] A. A. Razborov. Lower bounds on the monotone complexity of some boolean functions. *Soviet Math. Dokl*, 31:354–357, 1985.

[13] A. A. Razborov. On the method of approximations. In *Proc. 21st Annual ACM Symposium on the Theory of Computing*, pages 167–176, 1989.

[14] I. Wegener. *The Complexity of Boolean Functions.* B.G. Teubner and John Wiley & Sons, 1987.

[15] A. C. Yao. A lower bound for the monotone depth of connectivity. In *Proc. 35th Symposium on Foundations of Computer Science*, pages 302–308, 1994.