Electronic Colloquium on Computational Complexity, Report No. 17 (2011)

NEXP does not have non-uniform quasi-polynomial-size ACC circuits of $o(\log \log n)$ depth

Fengming Wang*
Department of Computer Science
Rutgers University
New Brunswick, NJ, 08855 USA
fengming@cs.rutgers.edu

February 1, 2011

Abstract

ACC_m circuits are circuits consisting of unbounded fan-in AND, OR and MOD_m gates and unary NOT gates, where m is a fixed integer. We show that there exists a language in non-deterministic exponential time which can not be computed by any non-uniform family of ACC_m circuits of quasi-polynomial size and $o(\log \log n)$ depth, where m is an arbitrarily chosen constant.

1 Introduction

Proving non-uniform circuit lower bounds is a longstanding open problem in complexity theory. The lack of progress in nearly two decades has made it a well-known major challenge in the theoretical computer science community. Recently, Williams [15] proposed a research program which tried to show circuit lower bounds via designing fast satisfiability algorithms for Circuit-SAT problems. A few months ago, Williams [16] succeeded in carrying out the program by proving an ingenious super-polynomial lower bound for NEXP against non-uniform constant-depth ACC circuits of polynomial size, thereby solving a notorious long-standing open problem. For more background and history on circuit lower bounds, we refer the readers to [16] which elaborates on this history in detail.

*Supported in part by NSF Grants CCF-0830133 and CCF-0832787.

ISSN 1433-8092

In this paper, we show that Williams' lower bound result can be extended to a broader class of ACC circuits with non-constant depth. A function f is a quasi-polynomial if $f = n^{\log^{O(1)} n}$. Formally, our main theorem is stated as:

Theorem 1 NEXP *does not have non-uniform ACC circuits of quasi-polynomial size and* $o(\log \log n)$ *depth.*

1.1 Related work

In this section, we survey a few examples of earlier work giving super-polynomial size bounds for circuits of non-constant depth.

More than two decades ago, building on his powerful switching lemma, Håstad [9] proved that the parity function can not be computed by families of AC circuits of polynomial size and depth at most $\frac{c \log n}{\log \log n}$ for some positive constant c. This result found many applications in proving lower bounds for the parallel random access machine model (PRAM), which is one of the widely adopted models of parallel computation. For instance, Beame and Håstad [5] exhibited the optimal $\Omega(\frac{\log n}{\log \log n})$ lower bounds on the time for CRCW (Concurrent read and concurrent write) PRAM with polynomially many processors to compute the parity function and related problems.

The classic results of Razborov [12] and Smolensky [13] showed that if p is a prime and q is not a power of p, then the MOD_q function is not computable any constant-depth and poly-size family of ACC_p circuits. In fact, their technique also works in the regime of non-constant-depth circuit lower bounds. More precisely, one can adapt their polynomial method to show that the same MOD_q function remains hard even for ACC_p circuits of polynomial size and $\Omega(\frac{\log n}{\log \log n})$ depth. Even though the results in this paper are exponentially worse in terms of circuit depth, note that they hold for the more powerful ACC circuit model.

Some other $\Omega(\log \log n)$ depth bounds are known in the setting of *uniform* circuits. Allender and Gore [2] showed that the permanent function is not computable by DLOGTIME-uniform ACC^0 circuits of exponential size. Later Allender [1] proved a smaller (but still super-quasi-polynomial) bound for computing the permanent on DLOGTIME-uniform *threshold* circuits .Koiran and Perifel [11] extended this latter result [1], and proved that the permanent function can not be computed by DLOGTIME-uniform threshold or arithmetic circuits of polynomial size and $o(\log \log n)$ depth.

2 Preliminaries

We assume that the readers are familiar with standard notations for complexity classes [3] and circuit complexity classes [14]. General circuits consist of NOT gates and unbounded fan-in AND and OR gates. ACC_m circuits are general circuits equipped with unbounded fan-in MOD_m gates, where m is a fixed integer.

We say a boolean function $g : \Sigma^n \to \{0,1\}$ is in $\text{ACC}_m(s,d)$ if g can be recognized by some ACC_m circuit of size at most s and depth bounded by d. For any two functions $s(n)$ and $d(n)$, we say a language $L \in \text{ACC}(s(n),d(n))$ if there exists an integer constant m such that for each input length n, its characteristic function L_n is in $\text{ACC}_m(s(n),d(n))$. For any two families of functions $\mathcal{S}$ and $\mathcal{D}$, $\text{ACC}(\mathcal{S},\mathcal{D}) = \bigcup \text{ACC}(s(n),d(n)) \mid s(n) \in \mathcal{S}, d(n) \in \mathcal{D}$.

SYM^+ circuits have exactly two levels of internal nodes. The top level is a single gate with unbounded fan-in which computes an arbitrary symmetric function and the bottom level contains only AND gates which are connected directly to the input variables. We say a boolean function $g : \Sigma^n \to \{0,1\}$ is in $\text{SYM}^+(s,t)$ if it can be computed by some SYM^+ circuit of size at most s, where moreover, the fan-in of AND gates is bounded by t. We can define similarly as above the language classes $\text{SYM}^+(s(n),t(n))$ and $\text{SYM}^+(\mathcal{S},\mathcal{T})$.

For a circuit type $\mathcal{C}$ and a set of associated measures, it will be convenient for us to consider the family of collections of boolean circuits which is denoted as $\text{Circuit}_{\mathcal{C}}(s_1(n),s_2(n),..,s_m(n)) = \{G_1,G_2,..\}$, where each circuit in G_n has exactly n input variables and its ith measure is bounded by $s_i(n)$ respectively. For general circuits, we only consider the size measure, hence, $\text{Circuit}_{\text{General}}(s(n)) = \{G_1,G_2,..\}$, where G_n contains all circuits of size at most $s(n)$. We can also give similar definitions for $\text{Circuit}_{\text{ACC}_m}(s(n),d(n))$ with both size and depth measures and for $\text{Circuit}_{\text{SYM}^+}(s(n),t(n))$ where the first measure is the size measure and the second measure is in terms of the bottom fan-in.

For two families $\mathcal{F}_1 = \{G_1,G_2,..\}$ and $\mathcal{F}_2 = \{G'_1,G'_2,..\}$, we say $\mathcal{F}_1$ is *transformable to* $\mathcal{F}_2$ if for all sufficiently large $n \in \mathbb{N}$, $\forall C \in G_n$, $\exists C' \in G'_n$ such that $\forall x \in \Sigma^n, C(x) = C'(x)$, namely, C and C' are equivalent. Furthermore, $\mathcal{F}_1$ is *transformable to* $\mathcal{F}_2$ *in time* $t(n)$ if there exists a uniform algorithm which given the standard encoding of C, output C' in time $t(n)$.

For a family $\mathcal{F} = \{G_1,G_2,..\}$, we say $\mathcal{F}$-SAT is *solvable in time* $t(n)$ if there exists a uniform algorithm $\mathcal{A}$ such that for all sufficiently large $n \in \mathbb{N}$, given an arbitrary C in G_n, A decides its satisfiability in time $t(n)$.

3 Main result

3.1 A fast satisfiability algorithm

Transformation between different circuit types is an important building block in our proof. Yao [17], Beigel and Tarui [7] and Allender and Gore [2] studied conversion from Circuit$_{\text{ACC}_m}(n^{O(1)}, O(1))$ to Circuit$_{\text{SYM}^+}(n^{\log^{O(1)} n}, \log^{O(1)} n)$. In fact, their strategy works in a more general setting.

Fix m to be an integer constant.

Theorem 2 ([7, 2]) *There is a universal constant c such that for any size function $s(n)$ and any depth function $d(n)$, the family ACC$_m(s(n), d(n))$ is transformable in time $2^{O((\log s(n))^{2^{cd(n)}})}$ to the family SYM$^+(2^{(\log s(n))^{2^{cd(n)}}}, (\log s(n))^{2^{cd(n)}})$.*

Corollary 3 *For any small constant ϵ, any quasi-polynomial $p(n)$ and any depth function $d(n)$ of order $o(\log\log n)$, the family ACC$_m(p(n), d(n))$ is transformable to the family SYM$^+(2^{n^\epsilon}, n^\epsilon)$ in time $2^{O(n^\epsilon)}$.*

In [16], Williams gave an algorithm for solving the satisfiability problem of SYM$^+$ circuits of size s over n variables in time $O((2^n + s)n^{O(1)})$. Combining it with Corollary 3, the following theorem is immediate.

Theorem 4 *There exists a constant c such that for any quasi-polynomial $p(n)$ and any depth function $d(n)$ of order $o(\log\log n)$, Circuit$_{ACC_m}(p(n), d(n))$-SAT is solvable in time $O(2^n n^c)$.*

The running time above can indeed be improved.

Theorem 5 *For any positive constant c', any quasi-polynomial $p(n)$ and any depth function $d(n)$ of order $o(\log\log n)$, Circuit$_{ACC_m}(p(n), d(n))$-SAT is solvable in time $O(\frac{2^n}{n^{c'}})$.*

Proof: Let c be the constant in Theorem 4. Given an ACC$_m(p(n), d(n)$ circuit over n variables, when the first $(c + c')\log n$ inputs are set to definite values, we simplify it to obtain a circuit over $n - (c + c')\log n$ many variables. Hence, by fixing the first $(c + c')\log n$ input variables to all possible sequences, we get $n^{c+c'}$ many circuits. Create a new circuit by feeding their outputs to a single OR gate. The size of this new circuit is bounded by $p(n)n^{c+c'}$ and its depth is only increased by one. Note that $p(n)n^{c+c'}$ is still a quasi-polynomial in $(n - (c + c')\log n)$, and $d(n) + 1$ is in $o(\log\log(n - (c+c')\log n))$ as well. By Theorem 4, its satisfiability can be determined in time $O(2^{n-(c+c')\log n}(n - (c + c')\log n)^c)$ which is $O(\frac{2^n}{n^{c'}})$.

This finishes our arguments since the satisfiability problem for the new circuit is equivalent to the one for the original circuit. □

Note: The above strategy is very similar to the one adopted by [15], where about n^{δ} many input variables are set in a single copy. However, it is crucial for our work to keep the size of the final circuit within quasi-polynomial (compared to $2^{n^{O(\delta)}}$ in [15]) in order to apply Theorem 4.

3.2 Proof of main theorem

In this section, we present our main lower bound result via the framework invented by Williams [16]. The following notions will be useful.

Definition 1 *Let $x = x_0x_1x_2...x_{|x|-1}$ be a binary string, where $|x|$ is the size of x. We say x is* succinctly represented by the circuit C *if C has $\lceil \log(|x|+1) \rceil$ many input bits and moreover, for all $0 \leq i \leq |x|-1$, $C(i) = x_i$ while its output can be arbitrary otherwise. We call such a circuit C as a* succinct representations *of x.*

Let ϕ be a 3-CNF formula with n variables and m clauses. ϕ is succinctly represented by the circuit C' *if C' has $\lceil \log(m+1) \rceil$ many input bits and furthermore, on the input $0 \leq i \leq m-1$, $C'(i)$'s output is the standard binary encoding of the ith clause. Hence, C' has roughly $3(\lceil \log(n+1) \rceil + 1)$ output bits, the amount which is needed to encode three literals. We say that C' is a* succinct representation or compression *of ϕ.*

Theorem 6 (Theorem 1 restated) NEXP $\not\subseteq$ *ACC*$(n^{\log^{O(1)} n}, o(\log\log n))$.

Proof: Suppose NEXP $\subseteq$ ACC$(n^{\log^{O(1)} n}, o(\log\log n))$. The first step of our proof is to note that, because of Theorem 5, it is possible to state a slight variant of Lemma 3.1 of [16].

Lemma 7 *There is a universal positive constant c with the following property. Assume that* P $\subseteq$ *ACC*$(n^{\log^{O(1)} n}, o(\log\log n))$*, then for every $L \in$* NTime$[2^n]$*, there is a nondeterministic algorithm $\mathcal{A}$, an integer constant m, a quasi-polynomial $p'(n)$ and a depth function $d'(n)$ of order $o(\log\log n)$ such that*

- *$\mathcal{A}$ runs in $O(\frac{2^n}{n^{c'}})$ time,*
- *for every instance x with $|x| = n$, $\mathcal{A}(x)$ either rejects or prints a circuit $C_x \in G_{n+c\log n}$ where $G_{n+c\log n} \in$ Circuit$_{ACC_m}(p'(n), d'(n))$ such that $x \in L$ if and only if C_x is the compression of a satisfiable 3-CNF formula F_x of size $2^n \cdot n^{O(1)}$, and*

- *there is at least one computation path $\mathcal{A}(x)$ that outputs C_x.*

Hence, Lemma 7 implies that as long as deciding the satisfiability of succinct 3-CNF instances such as C_x can be achieved in nondeterministic time $O(\frac{2^n}{n^{c'}})$ for any c', then NTime$[2^n] \subseteq$ NTime$[\frac{2^n}{n^{c'}}]$, in contradiction to the nondeterministic time hierarchy [18]. Therefore, we are done except for showing that the satisfiability of C_x can be tested in this time bound, assuming that NEXP $\subseteq$ ACC$(n^{\log^{O(1)} n}, o(\log\log n))$.

The following theorem is a variant of Theorem 5.2 in [16]. It is also implicit in the work of Impagliazzo, Kabanets and Wigderson [10].

Theorem 8 ([10, 16]) NEXP $\subseteq$ SIZE$(n^{\log^{O(1)} n})$ *implies that for every language L in* NEXP*, there exists a quasi-polynomial p such that $\forall x \in L$, there exists a witness w for x with the property that the boolean function whose truth table is given by w can be computed by a general circuit of size at most $p(|x|)$.*

In other words, every instance in L has a succinctly represented witness. In particular, every compressed 3*-CNF formula has a succinct satisfying assignment since the Succinct SAT Problem is in* NEXP*.*

Our assumption that NEXP $\subseteq$ ACC$(n^{\log^{O(1)} n}, o(\log\log n))$ implies NEXP $\subseteq$ SIZE$(n^{\log^{O(1)} n})$, so obviously the conclusion in Theorem 8 holds.

Lemma 9 (Folklore) *If* P $\subseteq$ *ACC$(n^{\log^{O(1)} n}, o(\log\log n))$, then there exists a universal constant m' such that for any quasi-polynomial $p(n)$, there exists a quasi-polynomial $p'(n)$ and a depth function $d(n)$ of order $o(\log\log n)$ such that Circuit$_{General}(p(n))$ is transformable to Circuit$_{ACC_{m'}}(p'(n), d(n))$.*

Proof: The Circuit Value Problem (CVP) is in P, and hence, there exists an integer constant m', a quasi-polynomial $q(n)$ and a depth function $d'(n) = o(\log\log n)$ such that CVP is computed by a family of ACC$_{m'}$ circuits of size at most $q(n)$ and depth bounded by $d'(n)$. Under the standard encoding of circuits, this implies that any general circuit of size at most $p(n)$ has an equivalent ACC$_{m'}$ circuit of size at most $q(p^2(n))$ and depth bounded by $d'(p^2(n))$. Since $q(p^2(n))$ is still a quasi-polynomial in n and $d'(p^2(n)) = o(\log\log n)$, our claim holds. □

Theorem 8 tells us that for every x in L, there exists a witness w that is succinctly represented by a circuit of quasi-polynomial size. By Lemma 9, this circuit can be assumed to be a quasi-polynomial-size ACC$'_m$ circuit C_w of depth $o(\log\log n)$. Thus analogous to the work of Williams [16], our algorithm for deciding the satisfiability of the succinctly represented 3-CNF instance C_x proceeds as the following steps.

1. Guess the circuit C_w of quasi-polynomial size and depth $o(\log \log n)$, where w is a witness for C_x being satisfiable.

2. Build a circuit C of the following form: On input i, use C_x to obtain the encoding of the ith clause of the formula F_x. Querying C_w, find the values of the three variables occurring in this clause, according to the witness w.

3. C rejects if and only if these values cause cause the clause to evaluate to 1.

Note that C is unsatisfiable if and only if every clause of F_x is satisfied by w.

Fact 10 *For two fixed integers m and m', there exists a polynomial r such that any ACC circuit containing both MOD$_m$ and MOD$_{m'}$ gates of size at most s can be simulated uniformly by an ACC$_l$ circuit of the same depth and size at most $r(s)$ where $l = m \cdot m'$.*

By fact 10, C is a quasi-poly-size ACC$_l$ circuit of depth at most $d(n) + d'(n) + O(1)$ and by Theorem 5, its satisfiability is decidable in time $O(\frac{2^n}{n^{c'}})$ for any c', which concludes our proof for the main theorem. □

4 Discussions

We have not fully exploited the strength of the machinery behind Theorem 2. The original form of the transformation provides a large set of parameters which can be tuned smoothly. For instance, one can allow $m(n) = \{l_1, l_2, ...\}$ to be a slowly growing (say, of order $O(\log \log n)$) integer sequence rather than a fixed constant and consider the circuit families ACC$_{m(n)}$ where the nth circuit contains the presence of MOD$_{l_i}$ gates for all $i \leq n$. It is easy for the readers who are familiar with the framework of [17], [7] and [2] to verify that NEXP does not have non-uniform ACC$_{m(n)}$ circuits of quasi-polynomial size and non-constant depth. This phenomenon has been observed by several authors, [4], [8] etc, and their further investigations made it explicit that SYM$^+(n^{\log^{O(1)} n}, \log^{O(1)} n)$ actually encompasses a circuit complexity class presumably larger than ACC$(n^{O(1)}, O(1))$, where every ACC$(n^{O(1)}, O(1))$ circuit has an extra symmetric gate at the top. Hence, it is natural to conjecture that NEXP is not contained in this class either. However, the proof of Theorem 5 introduces too many duplicate symmetric gates, which falls beyond the reach of current techniques. Note that Beigel [6] showed that polylog majority gates can be merged into one at the top, but his results would yield the trivial bound 2^{n^c} for some $c > 1$ in our case.

We would like to draw the comparison between this work and [13]. The main technical difficulty which prevents us from obtaining a depth lower bound of order $\Omega(\frac{\log n}{\log \log n})$ is that each application of modulus-amplifying polynomials creates extra AND gates of large fan-in. This in turn causes the snowball effect of the blow-up in the final circuit size. Thus, new ideas are needed in order to improve the current depth lower bound.

5 Acknowledgement

We thank Eric Allender, Luke Friedman and Ryan Williams for helpful discussions. We are especially grateful to Eric Allender for carefully reading earlier drafts and providing many useful comments.

References

[1] Eric Allender. The permanent requires large uniform threshold circuits. *Chicago J. Theor. Comput. Sci.*, 1999, 1999.

[2] Eric Allender and Vivek Gore. A uniform circuit lower bound for the permanent. *SIAM Journal on Computing*, 23(5):1026–1049, 1994.

[3] Sanjeev Arora and Boaz Barak. *Computational Complexity, a modern approach*. Cambridge University Press, 2009.

[4] David A. Mix Barrington. Quasipolynomial size circuit classes. In *Proc. IEEE Conf. on Structure in Complexity Theory*, pages 86–93, 1992.

[5] Paul Beame and Johan Håstad. Optimal bounds for decision problems on the crcw pram. *Journal of the ACM*, 36(3):643–670, 1989.

[6] Richard Beigel. When do extra majority gates help? polylog() majority gates are equivalent to one. *Computational Complexity*, 4:314–324, 1994.

[7] Richard Beigel and Jun Tarui. On ACC. *Computational Complexity*, 4:350–366, 1994.

[8] Richard Beigel, Jun Tarui, and Seinosuke Toda. On probabilistic acc circuits with an exact-threshold output gate. In *International Symposium on Algorithms and Computation*, pages 420–429, 1992.

[9] Johan Håstad. Almost optimal lower bounds for small depth circuits. In *Proc. ACM Symp. on Theory of Computing (STOC)*, pages 6–20, 1986.

[10] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. *Journal of Computer and System Sciences*, 65(4):672–694, 2002.

[11] Pascal Koiran and Sylvain Perifel. A superpolynomial lower bound on the size of uniform non-constant-depth threshold circuits for the permanent. In *Computational Complexity*, pages 35–40, 2009.

[12] Alexander Razborov. Lower bounds on the size of bounded-depth networks over a complete basis with logical addition. *Mathematical Notes of the Academy of Sciences. of the USSR*, 41(4):333–338, 1987.

[13] Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proc. ACM Symp. on Theory of Computing (STOC)*, pages 77–82, 1987.

[14] Heribert Vollmer. *Introduction to Circuit Complexity*. Springer, 1999.

[15] Ryan Williams. Improving exhaustive search implies superpolynomial lower bounds. In *Proc. ACM Symp. on Theory of Computing (STOC)*, pages 231–240, 2010.

[16] Ryan Williams. Non-uniform ACC circuit lower bounds. 2010.

[17] Andrew Chi-Chih Yao. On ACC and threshold circuits. In *Proc. IEEE Symp. on Found. of Comp. Sci. (FOCS)*, pages 619–627, 1990.

[18] Stanislav Zak. A turing machine hierarchy. *Theoretical Computer Science*, 26:327–333, 1983.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de