Circuit Lower Bounds for Average MA *

Alexander Knop
aaknop@gmail.com
Steklov Institute of Mathematics at St. Petersburg,
27 Fontanka, St.Petersburg, 191023, Russia

Abstract. Santhanam (2007) proved that MA/1 does not have circuits of size $n^k$. We translate his result to the average case setting by proving that there is a constant $a$ such that for any $k$, there is a language in AvgMA that cannot be solved by circuits of size $n^k$ on more than the $1 - \frac{1}{n^a}$ fraction of inputs.

In order to get rid of the non-uniform advice, we supply the inputs with the probability threshold that we use to determine the acceptance. This technique was used by Pervyshev (2007) for proving a time hierarchy for heuristic computations.

1 Introduction

A widely known counting argument shows that there are Boolean functions that have no polynomial-size circuits. However, all attempts to prove a superpolynomial lower bound for an explicit function (that is, function in NP) failed so far.

This challenging problem was attacked in three directions. The most obvious direction to prove weak lower bounds for specific functions did not yield anything better than the bound $3n - o(n)$ [Blu83] (the bound was improved to $5n - o(n)$ for circuits in de Morgan basis [ILMR02]). Another direction, to prove strong lower bounds for restricted classes of circuits yielded exponential bounds for monotone [Raz85] and bounded-depth circuits [Ajt83,Hås86], but did not attain superpolynomial bounds for circuits without such restrictions, and even for de Morgan formulas (of unrestricted depth).

Yet another way is to prove lower bounds for smaller and smaller complexity classes (aiming at NP). The exponential lower bound obtained by counting needs doubly exponential time. Buhrman et al. [BFT98] showed that it can be also done in MAEXP. A less ambitious goal is to prove lower bounds of the form $n^k$ (for each $k$), called fixed-polynomial lower bounds. This line of research was started by Kannan [Kan82] who showed that for each $k$ there is a language in $\Sigma_2P \cap \Pi_2P$ that has no circuits of size $n^k$. This was pushed down to $S_2P$ [Cai01]. However, attempts to push it down further to MA ended up in lower bounds for the classes PromiseMA, MA/1 [San07], which are not “normal” classes in the sense that PromiseMA is not a class of languages, and MA/1 is not a uniform class.

The obstacle that prevents proving the result for MA is typical for proving structural results (hierarchy theorems, the existence of complete problems) for semantic classes: Santhanam’s construction does not always satisfy the bounded-error condition (the promise) of MA. A similar obstacle was overcome by Pervyshev [Per07] for a hierarchy theorem for heuristic bounded-error randomized computations and many other heuristic classes and by Itsykson [Its09] for the existence of a AvgBPP-complete problem (though the existence of AvgMA-complete problems remained open).

In this paper we translate Santhanam’s result to the heuristic setting. Namely, we prove fixed-polynomial circuit lower bounds for AvgMA: there is a number $a > 0$ such that for every $k$, there exists a language $L$ such that

* The research is partially supported by the RFBR grant 14-01-00545, by the President’s grant MK-2813.2014.1, by the Government of the Russia (grant 14.Z50.31.0030), and by the Ministry of Education and Science of the Russian Federation, project 8216. The author is also supported by a fellowship from the Computer Science Center (St.Petersburg).
(1) there is a average polynomial-time Merlin-Arthur protocol for solving $L$ under the uniform distribution on the inputs, i.e., a Merlin-Arthur protocol that gets a confidence parameter $\delta$, runs in time polynomial in $\delta^{-1}$ and the size of the input, and correctly (with bounded probability of error) accepts or rejects a fraction $1 - \delta$ of the inputs and with high probability return failure on all other inputs;

(2) no $n^q$-size circuit can solve $L$ on more than a fraction $1 - \frac{1}{n^d}$ of the inputs.

Similarly to Santhanam’s proof, our proof consists of two parts. The easier part is conditioned on $\text{PSPACE} \subseteq \text{P/poly}$, and it follows from the resulting collapses. The main part is the construction of a hard language based on the assumption $\text{PSPACE} \not\subseteq \text{P/poly}$. In order to get rid of the non-uniform advice, we supply the inputs with the probability threshold that we use to determine the acceptance. (This technique was used by Pervyshev [Per07] for proving a time hierarchy for heuristic computations.) It follows that the fraction of the resulting inputs that have a “bad” threshold is small.

Organization of the paper. In Sect. 2 we give the definitions and recall the necessary background results. In Sect. 3 we prove the main result and show possible way to improve this result. Also we show some problems which occurs in this way.

2 Definitions

We first introduce some notation.

For two sets $S_1, S_2 \subseteq \{0, 1\}^n$ denote $\Delta(S_1, S_2) = \frac{|(S_1 \cup S_2) \setminus (S_1 \cap S_2)|}{2^n}$.

For language $L \subseteq \{0, 1\}^*$, denote $L^{=n} = L \cap \{0, 1\}^n$.

The characteristic function of $L$ is denoted by $L(x)$.

The main idea of the proof of our result is to take a hard language that is self-correctable and instance-checkable, and turn it into a language that has an AvgMA protocol while remaining sufficiently complex on the average. The self-correctness property is needed to convert a worst-case hard function into a function that is hard on the average. The instance checkability is needed to design a Merlin-Arthur protocol (where Arthur simulates the instance checker and Merlin sends a circuit family computing the oracle). We now formally define these two properties.

**Definition 1 ([TV02]).** Let $b \in \mathbb{Q}_+$. A language $L$ is $b$-self-correctable if there is a probabilistic polynomial-time oracle algorithm $A$ (self-corrector for $L$) such that for all languages $L'$ if $\Delta(L^{=n}, L'^{=n}) < \frac{1}{4^n}$, then $\forall x \in \{0, 1\}^n$, $\text{Pr}[A^{L'=f(n)}(x) = L(x)] > \frac{3}{4}$. We call a language self-correctable if it is $b$-self-correctable for some constant $b$.

This definition informally means that if we have oracle access to language that is close enough to $L$ then we can probabilistically decide $L$ in polynomial time.

**Definition 2 ([TV02]).** A language $L$ is $f$-instance-checkable if there is a probabilistic polynomial-time oracle algorithm $M$ (instance checker for $L$) such that for all $x \in \{0, 1\}^n$:

- if $x \in L$ then $\text{Pr}[M^{L=f(n)}(x) = 1] = 1$ (perfect completeness);
- if $x \notin L$ then for all $L'$ the following holds $\text{Pr}[M^{L'=f(n)}(x) = 1] < \frac{1}{2^n}$ (correctness).

**Definition 3.** Denote by $U$ the ensemble of uniform distributions on $\{0, 1\}^n$ (if $|x| = n$ then $U_n(x) = \frac{1}{2^n}$).

Also we need definition of classes languages which decided by circuits.

**Definition 4.**

1. Language $L$ contains in $\text{Size}[f(n)]$ iff there is family of circuits $C_n$ such that $|C_n| < f(n)$ and $C_{|x|}(x) = L(x)$.

2. Language $L$ contains in $\text{BPSize}[f(n)]$ iff there is family of randomized circuits $C_n$ such that $|C_n| < f(n)$ and $\text{Pr}[C_{|x|}(x) = L(x)] > \frac{3}{4}$ (probability taken over the randomness of the $C_n$).
3. Language \( L \) contains in \( \text{Heur}_{\delta(n)} \text{Size}[f(n)] \) iff there is a family of circuits \( C_n \) such that \( |C_n| < f(n) \) and \( \Pr_{x \sim U_n} [C_{|x|}(x) = L(x)] \geq 1 - \delta(n) \).

4. Language \( L \) contains in \( \text{Heur}_{\delta(n)} \text{BPSize}[f(n)] \) iff there is a family of randomized circuits \( C_n \) such that \( |C_n| < f(n) \) and \( \Pr_{x \sim U_n} [\Pr[y: C_{|x|}(x) = L(x)] > \frac{2}{3}] \geq 1 - \delta(n) \) (inner probability taken over the randomness of the \( C_n \)).

**Lemma 1.** For all functions \( \delta: \mathbb{N} \to [0; 1] \) and \( t: \mathbb{N} \to \mathbb{N} \),

\[
\text{Heur}_{\delta(n)} \text{BPSize}[t(n)] \subseteq \text{Heur}_{\delta(n)} \text{Size}[\text{poly}(t)(n)].
\]

**Proof.** A trivial extension of Adleman’s theorem (\( \text{BPP} \subseteq \text{P/poly} \)) yields the result. \( \square \)

**Lemma 2.** If language \( L \) is 1-self-correctable and \( L \in \text{Heur}_{1-\frac{1}{n}} \text{Size}[f(n)] \), then \( L \in \text{Size}[f(n)\text{poly}(n)] \).

**Proof.** We apply the standard transformation of a Turing machine that computes the self-corrector of \( L \) to a randomized circuit \( B \). We assume that instead of making oracle requests \( B \) uses a circuit that heuristically computes \( L \). Hence \( L \in \text{BPSize}[f(n)\text{poly}(n)] \) and \( L \in \text{Size}[f(n)\text{poly}(n)] \) by Lemma 1. \( \square \)

Classes \( \text{AvgC} \) make a errorless and “uniform” version of classes \( \text{Heur}_{\delta(n)} \text{C} \); namely, the “confidence” parameter \( \delta(n) \) is given to the decision algorithm as part of the input, and the algorithm is required to work in polynomial time both in the input size and \( \delta(n)^{-1} \). For clarity, we give the definition for the specific case of Merlin-Arthur protocols.

**Definition 5.** A language \( L \) has a heuristic Merlin-Arthur protocol (in short \( L \in \text{HeurMA} \)) iff there is a probabilistic algorithm \( A(x, y, \delta) \) (here \( x \) is the input, \( y \) is Merlin’s proof, and \( \delta \) is the confidence parameter) and a family of sets \( \{S_n^\delta \subseteq \{0, 1\}^n \}_{\delta \in \mathbb{Q}, n \in \mathbb{N}} \) (large sets of inputs where the protocol behaves correctly) such that for all \( n \) and \( \delta \),

- \( U_n(S_n^\delta) \geq 1 - \delta \),
- \( A(x, y, \delta) \) runs in time \( \text{poly}(\frac{2}{\delta}) \), and
- for every \( x \) in \( S_n^\delta \):
  \[
  x \in L \Rightarrow \exists y \Pr[A(x, y, \delta) = 1] > \frac{2}{3},
  \]
  \[
  x \notin L \Rightarrow \forall y \Pr[A(x, y, \delta) = 0] > \frac{2}{3}.
  \]

A language \( L \) has an average-case Merlin-Arthur protocol (in short \( L \in \text{AvgMA} \)) if in addition the following holds: for all \( x \) not in \( S_n^\delta \), then our protocol does not give wrong answer:

\[
  x \in L \Rightarrow \exists y \Pr[A(x, y, \delta) = 1] > \frac{2}{3} \lor \Pr[A(x, y, \delta) = \bot] > \frac{1}{6},
  \]
\[
  x \notin L \Rightarrow \forall y \Pr[A(x, y, \delta) = 0] > \frac{2}{3} \lor \Pr[A(x, y, \delta) = \bot] > \frac{1}{6}.
  \]

For the first case of our proof we need a \( \text{PSPACE} \) language with high heuristic circuit complexity (a collapse will put it into \( \text{MA} \)).

**Lemma 3 ([San07]).** There is a constant a such that for all \( k \),

\[
\text{PSPACE} \not\subseteq \text{Heur}_{1-\frac{1}{n^k}} \text{Size}[n^k].
\]

For the second case we need a \( \text{PSPACE} \)-complete language with good properties.

**Lemma 4 ([San07]).** There exists a \( \text{PSPACE} \)-complete language that is self-correctable and \( n \)-instance-checkable.
We need reductions somewhat similar yet different from randomized heuristic search reductions [BT06]: we do not need polynomial-time computability of the reduction (we will formulate a specific complexity requirement when needed), the disjointness of its images for different random strings and the uniformness of the distribution for each input.

**Definition 6.** Let $L$ and $L'$ be two languages, and $c: \mathbb{N} \to \mathbb{R}$ be a function. A collection of functions $f_n: \{0,1\}^n \times \{0,1\}^{y_n} \to \{0,1\}^{m_n}$ where $m_n \geq n$ is called a $c(n)$-heuristic reduction of $L$ to $L'$ if for all $x (|x| = n)$,
\[
x \in L \Rightarrow \forall r \in \{0,1\}^{y_n} \ f_n(x,r) \in L',
\]
\[
x \not\in L \Rightarrow \forall r \in \{0,1\}^{y_n} \ f_n(x,r) \not\in L',
\]
and
\[
\forall n \forall S \subseteq \{0,1\}^n \times \{0,1\}^{y_n} \ |f_n(S)| > c(n) \frac{|S|}{2^{m_n}}
\]

**Lemma 5.** For all $a$ if $L' \in \text{Heur}_1 - \frac{1}{a} \cdot \text{Size}[p(n)]$ and there is a $\frac{a}{n}$-heuristic reduction of $L$ to $L'$ computable by circuits of size $q(n)$, then $L \in \text{Heur}_1 - \frac{1}{a} \cdot \text{Size}[p(m_n) + q(n)\text{poly}(n)]$ (where $m_n$ is as in Definition 6 and $d$ is a constant).

**Proof.** Let $D_n$ be a $q(n)$-size circuit that computes the reduction $f_n$, and let $C_n$ be a circuit that decides $L'=n$ with error $\frac{1}{a+1}$. By Lemma 1 it suffices to prove that for sufficiently large $n$, $\Pr_C[|C(D(x,r)) \neq L(x)| \geq \frac{1}{2}] < \frac{1}{a^2}$ (here and in what follows $C$ and $D$ stands for $C_n$ and $D_n$ for appropriate $n$). Assume the contrary. Then
\[
\frac{|\{(x,r)|C(D(x,r)) \neq L(x)|\}}{2^{m_n+g_n}} \geq \frac{1}{4n^2}.
\]
However, using the correctness and the domination conditions we get
\[
\frac{|\{(x,r)|C(D(x,r)) \neq L'(x)|\}}{2^{m_n}} \geq \frac{|\{(x,r)|C(D(x,r)) \neq L'(D(x,r))|\}}{2^{m_n}} \geq \frac{|\{(x,r)|C(D(x,r)) \neq L(x)|\}}{2^{m_n}} \geq \frac{d}{n^d \cdot 4n^{a+l}} \geq \frac{1}{n^{a+l+1}},
\]
which contradicts the assumption on $C$. \qed

## 3 Lower bounds for HeurMA

In order to work in the heuristic setting, we need to pay the attention to the probabilities of the inputs. Because of that, we need a function that encodes triples without increasing the length too much.

**Definition 7.** Denote by $\langle \cdot, \cdot, \cdot \rangle$ the function from $\{0,1\}^n \times \{0,1\}^{g(n)} \times \{0,1\}^{y_n}$ to $\{0,1\}^{2 \log(n) + n + g(n) + 2}$ defined by $\langle x, p, z \rangle = \hat{a}x_1x_2\ldots x_{2^{g(n)}}z_1z_2\ldots z_{2^{g(n)}}$ and $g$ is a polynomial.

**Lemma 6.** For all polynomials $g$, $f$, integer $k$ and randomized algorithm $A$ that receive parameters $x$, $y$, $z$ and use $g(|x|)$ random bits, for the following language
\[
L = \{ \langle x,p,z \rangle | |p| = g(|x|), \exists C \Pr[A(x,C,z) = 1] \geq 0.9 \land |C| < f(|x|,|z|) \}
\]
we have that $L \in \text{AvgMA}$ (hence $L \in \text{HeurMA}$).

**Proof.** Consider the following protocol showing that $L \in \text{AvgMA}$.
1. Receive $C$ from Merlin.
   If $|z| > f(|x|, |z|)$ return 0.
2. If $\delta > \frac{1}{8^{2n}}$ then
   (a) Run \( \frac{16}{75} \) times \( A(x, C, z) \), calculate the fraction \( q \) of accepts.
   (b) If \( q \geq 0.75 / 4 \) then return 1;
   (c) if \( q \leq 0.6 / 7 \) then return 0;
   (d) else return \( \perp \).
3. If $\delta \leq \frac{1}{8^{2n}}$ then
   (a) Evaluate $q = \Pr[A(x, C, z) = 1]$ by running $A(x, C, z)$ on all possible random bits.
   (b) If $q \geq 0.75 / 4$ then return 1 else return 0.

Let us show that the size of the set $S_0^n$ where the protocol succeeds is large enough. If $\delta \leq \frac{1}{8^{2n}}$, the protocol always works correctly. Otherwise put $S_0^n = \{ (x, p, z) \in \{0, 1\}^n \mid |q(x, z) - 0.75/4| > \frac{1}{27} \}$ (note that $S_0^n \geq 1 - \delta$), where $q(x, z) = \max_x \Pr[A(x, C, z) = 1]$. Let us $q(x, C, z) = \Pr[A(x, C, z) = 1]$. If $x \in S_0^n$ then consider the following cases:

1. $\langle x, p, z \rangle \in L$ if Merlin sends $C$ such that $\Pr[A(x, C, z) = 1] > 0.75 / 4$.
   Then by Chernoff bound Arthur rejects with probability $\Pr[q < 0.75 / 4 < 2e^{-\frac{2 \delta}{4} \frac{n}{\log n}} = 2e^{-8} < \frac{1}{3}$;
2. $\langle x, p, z \rangle \not\in L$ for all $C$ we have that $\Pr[A(x, C, Z)] < 0.75 / 4$, hence by Chernoff bound Arthur accepts with probability $\Pr[q > 0.75 / 4 < 2e^{-\frac{2 \delta}{4} \frac{n}{\log n}} = 2e^{-8} < \frac{1}{3}$.

Otherwise if $x \not\in S_0^n$ then consider the following cases:

1. $\langle x, p, z \rangle \in L$ if Merlin sends $C$ such that $\Pr[A(x, C, z) = 1] > 0.75 / 4$.
   Then by Chernoff bound Arthur rejects with probability $\Pr[q \leq 0.75 / 4 < 2e^{-8} < \frac{1}{6}$, hence if Arthur accepts with probability $\Pr[q \geq 0.75 / 4 < 2e^{-8} < \frac{1}{6}$, then Arthur returns $\perp$ with probability $\Pr[q - 0.75 / 4 < 0] > \frac{1}{6}$;
2. $\langle x, p, z \rangle \not\in L$ for all $C$ we have that $\Pr[A(x, C, Z)] = 1$, hence by Chernoff bound Arthur accepts with probability $\Pr[q > 0.75 / 4 < 2e^{-8} < \frac{1}{6}$, hence if Arthur rejects with probability $\Pr[q > 0.75 / 4 < 2e^{-8} < \frac{1}{6}$, then Arthur returns $\perp$ with probability $\Pr[q - 0.75 / 4 < 0] > \frac{1}{6}$.

\[ \square \]

Lemma 7. If \( \text{PSPACE} \subseteq \text{P/poly} \) then there is constant $a > 0$ such that for all $k$ we have that $\text{MA} \not\subseteq \text{Heur}_{1− \frac{1}{a}} \text{Size}[n^k]$.

Proof. It is well known that from $\text{PSPACE} \subseteq \text{P/poly}$ follows that $\text{MA} = \text{PSPACE}$ (because the prover in the interactive protocol for $\text{QBF}$ [Sha90] can be replaced by a family of circuits sent by Merlin). Then Lemma 3 gives a language in $\text{MA}$ that has high heuristic complexity w.r.t. the uniform distribution. \[ \square \]

Theorem 1. There is a constant $a > 0$ such that for all $k \in \mathbb{Q}_+$,

\[ \text{AvgMA} \not\subseteq \text{Heur}_{1− \frac{1}{a}} \text{Size}[n^k]. \]

Proof. Let $L$ be as in Lemma 4 and $M$ be its instance checker (Def. 2). Fix any $k \in \mathbb{Q}_+$. Assume that $M$ uses $g(n)$ random bits for $n$-bit inputs. If $L \in \text{P/poly}$ then $\text{PSPACE} \subseteq \text{P/poly}$, and Lemma 6 implies desirable result.

Assume now that $L \not\in \text{P/poly}$. We will pad it to bring the language from $\text{PSPACE}$ down to polynomial complexity while keeping it above the complexity $n^k$. We will also supply the inputs with the number that we will use as the acceptance threshold for the instance checker. Namely, consider the language

\[ L' = \{ (x, p, z) \mid |p| = g(|x|), \exists \text{ circuit } C \ \Pr[M^C(x) = 1] \geq 0.75 / 4 \wedge |C| < (|z| + 1)^k + 1 \}. \]
Remark: Note that if we drop the requirement on the size of C, put \( p = 2^{g(|x|)} \) and let C be the circuit for L, then we will obtain a padded version of L (by perfect completeness of instance-checker).

It is easy to see that by Lemma 6 \( L' \in \text{AvgMA} \).

We now turn to proving that \( L' \not\subseteq \text{Heur}_{1-\frac{1}{2}}\text{Size}[n^k] \).

Let b be such a constant that L is b-self-correctable. Let \( a = b + 3 \). Assume, for the sake of contradiction, that \( L' \in \text{Heur}_{1-\frac{1}{2}}\text{Size}[n^k] \). Let \( s(n) \) be the worst-case circuit complexity of L and let \( y_n \) be such that \( y_n^{k+1} \leq s(n) < (y_n + 1)^{k+1} \). Consider \( f_n: \{0,1\}^n \times \{0,1\}^{g(n)+y_n^{-1}} \to \{0,1\}^{2\log(n)+2+n+g(n)+y_n} \) such that \( f_n(x,r_1,r_2) = (x,1r_1,r_2) \), where \( |r_1| = g(|x|) - 1 \) and \( |r_2| = y_n \). Let us prove that \( f_n \) is a \( \frac{1}{8n^2} \)-heuristic reduction from L to L'.

- The domination condition holds because the encodings of triplets form a \( \frac{1}{4n^2} \) fraction of the set of all strings and because we fix only the first bit in the second part of the triplet.
- The correctness condition is satisfied for \( x \in L \) since there is a circuit for L with size between \( y_n^{k+1} \) and \( (y_n + 1)^{k+1} \), hence by perfect completeness of instance checker for all \( r_1 \), \( (x,1r_1,r_2) \in L' \).
  For \( x \not\in L \), there are no circuits that force the instance checker to accept x with probability more than \( \frac{1}{2n^2} \) (note that by fixing the first bit of the second part of the triplet to 1 we require the probability more than \( \frac{1}{2} \)). Hence \( (x,1r_1,r_2) \not\in L' \).

So Lemma 5 for \( l = 2 \) and \( d = \frac{1}{2} \) implies \( L \in \text{Heur}_{1-\frac{1}{2}}\text{Size}[\{(y_n + g + 2\log(n) + n + 2)^k + (n + g + y_n + 2\log(n) + 2)\text{poly}(n)\}] \). Since L is b-self-correctable, by Lemma 2 we have \( L \subseteq \text{Size}[\{(y_n + g + 2\log(n) + 2)\text{poly}(n)\}] \subseteq \text{Size}[\text{poly}(n)] \). Hence \( y_n^{k+1} < s(n) < y_n^k \text{poly}(n) \) and hence \( y_n \) is bounded by polynomial therefore \( L \in \text{P/poly} \); contradiction with our assumption. \( \square \)

Further directions

All previous results in the same direction are closed under complement (for example, Santhanam’s lower bound [San07] for \( \text{MA}/1 \) is actually a lower bound for \( \text{(MA} \cap \text{co-MA})/1 \). It would be interesting to strengthen the result of this paper to a lower bound for \( \text{HeurMA} \cap \text{Heur co-MA} \).

Another open question is to replace in Theorem 1 the error \( 1 - \frac{1}{2^a} \) by \( \frac{1}{2} + \frac{1}{2^a} \) (possibly for every \( a > 0 \)).

Switching to \( \text{AM} (= \text{BP} \cdot \text{NP}) \) and decreasing the number of random bits in the protocol would de-randomize Theorem 1 down to heuristic \( \text{NP} \) and lead consequently to the lower bound \( \text{NP} \not\subseteq \text{Size}[n^k] \) for classical computations. However, as shown in Section 3, this needs non-relativizable techniques.

Acknowledgement

The author is grateful to Edward A. Hirsch for bringing the problem to his attention, to Dmitry Itsykson and anonymous referees for their comments that significantly improved the (initially unreadable) presentation.

Bibliography

References


[San07] Santhanam. Lower bounds for \( \text{MA}/1 \) are closed under complement.