Electronic Colloquium on Computational Complexity, Report No. 37 (2013)

# Circuit Lower Bounds for Heuristic MA [⋆]

Alexander Knop
aaknop@gmail.com

Saint-Petersburg State University
28 University prospect, 198504, Stary Peterhof, St.Petersburg, Russia

**Abstract.** Santhanam (2007) proved that **MA**/1 does not have circuits of size $n^k$. We translate his result to the heuristic setting by proving that there is a constant $a$ such that for any $k$, there is a language in Heur**MA** 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]. 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).

A 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 $\mathbf{MA_{EXP}}$. A less ambitious goal is to prove lower bounds in 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 $\Sigma^2\mathbf{P} \cap \Pi^2\mathbf{P}$ that has no circuits of size $n^k$. This was pushed down to $S_2\mathbf{P}$ [Cai01]. However, attempts to push down it further to **MA** ended up in lower bounds for the classes Promise**MA**, **MA**/1 [San07], which are not “normal” classes in the sense that Promise**MA** 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 Heur**BPP**-complete problem (though the existence of Heur**MA**-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 Heur**MA**: there is a number $a > 0$ such that for every $k$, there exists a language $L$ such that

(1) there is a heuristic 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 the fraction $1 - \delta$ of the inputs;

[⋆] Supported 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).

ISSN 1433-8092

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

Similarly to Santhanam's proof, our proof consists of two parts. The easier part is conditioned on $\mathbf{PSPACE} \subseteq \mathbf{P/poly}$, and it follows from the resulting collapses. The main part is the construction of a hard language based on the assumption $\mathbf{PSPACE} \not\subseteq \mathbf{P/poly}$. 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.

## 2 Definitions

We first introduce some notation.

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

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 is to take a hard language that is self-correctable and instance-checkable, and turn it into a language that has a Heur**MA** 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 $b \in \mathbb{Q}_+$. We call language $L$ $b$-self-correctible if there is a probabilistic polynomial-time oracle algorithm $A$ (self-corrector for $L$) such that for all languages $L'$:*

$$\mathit{\Delta}(L^{=n}, L'^{=n}) < \frac{1}{n^b} \Rightarrow \forall x \in \{0,1\}^n,\ \Pr[A^{L'^{=n}}(x) = L(x)] > \frac{3}{4}.$$

*We call a language self-correctible if it is $b$-self-correctible for some constant $b$.*

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

$$x \in L \Rightarrow \Pr[M^{L^{=f(n)}}(x) = 1] = 1$$
$$x \notin L \Rightarrow \forall L' \ \Pr[M^{L'^{=f(n)}}(x) = 1] < \frac{1}{2^n}$$

In order to define heuristic computations, we need to define distributional problems. Although in our case we only use problems with the uniform distribution, we give the definition for the general case.

**Definition 3.** *A distributional problem is a pair $(L, D)$ that consists of a language $L$ and an ensemble of distributions $D = \{D_n\}_{n \in \mathbb{N}}$, where $D_n$ is a distribution over a finite set of bit strings.*

(Note that $D_n$ is not necessarily distributed on $\{0,1\}^n$.)

**Definition 4.** *Let $C$ be a class of languages. Then the class $\mathrm{Heur}_{f(n)}C$ of distributional problems contains a problem $(L, D)$ iff there is a language $L' \in C$ such that*

$$\forall n \Pr_{x \leftarrow D_n}[L(x) = L'(x)] \geq f(n).$$

Classes Heur$C$ make a "uniform" version of classes Heur$_{f(n)}C$: namely, the "confidence" parameter $f(n)$ is given to the decision algorithm on the input, and the algorithm is required to work in polynomial time both in the input size and $(1-f(n))^{-1}$. For clarity, we give the definition for the specific case of Merlin-Arthur protocols.

**Definition 5.** *A distributional problem $(L,D)$ has a heuristic Merlin-Arthur protocol (in short $(L,D) \in$ Heur**MA**) 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 $\{S^n_\delta\}_{\delta\in\mathbb{Q}_+, n\in\mathbb{N}}$ (large sets where the protocol behaves correctly) such that for all $n$ and $\delta$,*

- *$D_n(S^n_\delta) > 1-\delta$,*
- *$A(x,y,\delta)$ runs in time $\mathrm{poly}(\frac{n}{\delta})$, and*
- *for every $x$,*

$$x \in L \cap S^n_\delta \Rightarrow \exists y \ \Pr[A(x,y,\delta)=1] > \frac{2}{3}$$
$$x \in \overline{L} \cap S^n_\delta \Rightarrow \forall y \ \Pr[A(x,y,\delta)=1] < \frac{1}{3}.$$

**Definition 6.** *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 7.** *1. Language $L$ contains in* **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* **BPSize**$(f(n))$ *iff there is family of circuits $C_n$ such that $|C_n| < f(n)$ and $Pr[C_{|x|}(x) = L(x)] > \frac{3}{4}$.*

**Lemma 1.** *Let all functions $f\colon \mathbb{N} \to (0;1]$ and $t\colon \mathbb{N}\to\mathbb{N}$,*

$$\mathrm{Heur}_{f(n)}\mathbf{BPSize}(t(n)) \subseteq \mathrm{Heur}_{f(n)}\mathbf{Size}(poly(n)t(n)).$$

*Proof.* A trivial extension of Adleman's theorem (**BPP** $\subseteq$ **P/poly**) yields. □

**Lemma 2.** *If language $L$ is a-self-correctible and $(L,U) \in \mathrm{Heur}_{1-\frac{1}{n^a}}Size(f(n))$, then $L \in$* **Size**$(f(n)poly(n))$.

*Proof.* Obviously $L \in$ **BPSize**$(f(n)\mathrm{poly}(n))$ and by Lemma 1 $L \in$ **Size**$(f(n)\mathrm{poly}(n))$. □

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

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

$$\{(L,U) \mid L \in \mathbf{PSPACE}\} \not\subseteq \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}(n^k).$$

For the second case we need a **PSPACE**-complete language with good properties.

**Lemma 4 ([San07]).** *There exists a* **PSPACE***-complete language that is self-correctible 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 8.** *Let $L$ and $L'$ be two languages, and $c\colon \mathbb{N} \to \mathbb{R}$ be a function. A collection of functions $f_n\colon \{0,1\}^n \times \{0,1\}^{y_n} \to \{0,1\}^{m_n}$ is called a $c(n)$-heuristic reduction of $L$ to $L'$ if for all $x$ ($|x| = n$),*

$$\begin{array}{c} x \in L \Rightarrow \forall r \in \{0,1\}^{y_n}\ \ f_n(x,r) \in L', \\ x \notin L \Rightarrow \forall r \in \{0,1\}^{y_n}\ \ f_n(x,r) \notin L', \end{array} \qquad (\textit{correctness})$$

*and*

$$\forall n\ \forall S \subseteq \{0,1\}^n \times \{0,1\}^{y_n}\ \ \frac{|f_n(S)|}{2^{m_n}} > c(n)\frac{|S|}{2^{n+y_n}} \qquad (\textit{domination})$$

**Lemma 5.** *If $L' \in \mathrm{Heur}_{1-\frac{1}{n^{a+l+1}}}\mathbf{Size}(p(n))$ and there is a $\frac{d}{n^l}$-heuristic reduction of $L$ to $L'$ computable by circuits of size $q(n)$, then $L \in \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}((p(m_n) + q(n))poly(n))$ (where $m_n$ is as in Definition 8 and $d$ is a constant).*

*Proof.* Let $D_n$ be a size-$q(n)$ circuit that computes the reduction $f_n$, and let $C_n$ be a circuit that decides $L'^{=n}$ with error $\frac{1}{n^{a+l+1}}$. By Lemma 1 it suffices to prove that for sufficiently large $n$, $\Pr_x[\Pr_r[C(D(x,r)) \neq L(x)] > \frac{1}{4}] < \frac{1}{n^a}$ (here and in what follows $C$ and $D$ stands for $C_{n'}$ and $D_{n'}$ for appropriate $n'$). Assume the contrary. Then

$$\frac{|\{(x,r)|C(D(x,r)) \neq L(x)\}|}{2^{n+y_n}} \geq \frac{1}{4n^a}.$$

However, using the correctness and the domination conditions we get

$$\begin{aligned} \frac{|\{y|C(y) \neq L'(y)\}|}{2^{m_n}} \geq {} & \frac{|\{D(x,r)|C(D(x,r)) \neq L'(D(x,r))\}|}{2^{m_n}} = \text{(by correctness)} \\ & \frac{|\{D(x,r)|C(D(x,r)) \neq L(x))\}|}{2^{m_n}} \geq \text{(by domination)} \\ & \frac{d}{n^l}\frac{|\{(x,r)|C(D(x,r)) \neq L(x)\}|}{2^{n+y_n}} \geq \frac{d}{4n^{a+l}}, \end{aligned}$$

which contradicts the assumption on $C$. □

## 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 9.** *Denote by $\langle \cdot, \cdot, \cdot \rangle$ the function from $\{0,1\}^n \times \{0,1\}^{g(n)} \times \{0,1\}^{y_n}$ to $\{0,1\}^{2\log(n)+n+g(n)+y_n+2}$ defined by $\langle x,p,z \rangle = \widehat{n}11xpz$, where $\widehat{x_1x_2\ldots} = x_10x_20\ldots$ and $g$ is a polynomial.*

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

$$\mathrm{Heur}\mathbf{MA} \not\subseteq \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}(n^k)$$

*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 $L \in \mathbf{P/poly}$ then $\mathbf{PSPACE} \subset \mathbf{P/poly}$, and hence $\mathbf{MA} = \mathbf{PSPACE}$, because the prover in the interactive protocol for `QBF` [Sha90] can be replaced by a family of circuits sent by Merlin. Then Lemma 3 gives a language in $\mathbf{MA}$ that has high heuristic complexity w.r.t. the uniform distribution.

Assume now that $L \notin \mathbf{P/poly}$. We will pad it to bring the language from $\mathbf{PSPACE}$ 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

$$\begin{aligned} L' = \{\langle x,p,z \rangle |\ & p \leq 2^{g(|x|)}, |z| \text{ is a power of two}, \\ & \exists \text{ circuit } C\ \Pr[M^C(x) = 1] \geq \frac{p}{2^{g(|x|)}} \wedge \\ & \quad |z|^{k+1} \leq |C| < (2|z|)^{k+1}\}. \end{aligned}$$

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 $L$. Then it is clear that for some length of $z$ the size of this circuit will fall between the size constraints. However, for "wrong" circuit $C$, which does not compute $L$ (in particular, for "wrong" length of $z$), it is not clear how to check $\langle x, 2^{g(|x|)}, z\rangle$ without possibly violating the bounded-error promise. When we allow $p$ to be arbitrary, we allow to violate promise for *some* values of $p$ while satisfying the bounded-error requirement for most other values. This is formalized in the following protocol showing that $(L', U) \in \mathrm{Heur}\mathbf{MA}$.

```
1. Receive circuit $C$ from Merlin.
   If $|C| \notin [|z|^{k+1}, (2|z|)^{k+1})$ return 0.
2. If $\delta > \frac{1}{2^{g(n)}}$ then
   (a) Run $\frac{2}{\delta^2}$ times $M^C(x)$, calculate the fraction $\bar{q}$ of accepts.
   (b) If $\bar{q} \geq \frac{p}{2^{g(n)}}$ then return 1 else return 0.
3. If $\delta \leq \frac{1}{2^{g(n)}}$ then
   (a) Evaluate $q = \Pr[M^C(x) = 1]$ by running $M^C$ on all possible random bits.
   (b) If $q \geq \frac{p}{2^{g(n)}}$ then return 1 else return 0.
```

Let us show that the size of the set $S^n_\delta$ where the protocol succeeds is large enough. If $\delta \leq \frac{1}{2^{g(n)}}$, the protocol always works correctly. Otherwise put $S^n_\delta = \{\langle x, p, z\rangle \mid |q(x) - \frac{p}{2^{g(n)}}| > \frac{\delta}{2}\}$, where $q(x) = \max_C \Pr[M^C(x) = 1]$. If $\langle x, p, z\rangle \in L \cap S^n_\delta$, Merlin sends the correct $C$ (with highest acceptance probability) and if $\langle x, p, z\rangle \in \bar{L} \cap S^n_\delta$, we can assume that Merlin maximizes the probability of acceptance and sends us a circuit that maximizes $q$ (we may assume this because Merlin needs to convince us that the probability of acceptance is higher than $\frac{p}{2^{g(n)}}$). Then by Chernoff bound $\Pr[|\bar{q} - q(x)| \geq \frac{\delta}{2}] < 2e^{-\delta^2 \frac{2}{\delta^2}} = 2e^{-2}$. Hence if $|q(x) - \frac{p}{2^{g(n)}}| > \frac{\delta}{2}$, then the probability of error is less than $\frac{1}{3}$.

We now turn to proving that $(L', U) \notin \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}(n^k)$.

Let $b$ be the constant from the definition of a self-correctible **PSPACE**-complete language (Def. 1). Let $a = b + 3$. Assume, for the sake of contradiction, that $L' \in \mathrm{Heur}_{1-\frac{1}{n^a}}\mathbf{Size}(n^k)$. Let $s(n)$ be the worst-case circuit complexity of $L$ and let $y_n$ be such that $y_n^{k+1} \leq s(n) < (2y_n)^{k+1}$. Consider $f_n(x, r_1 r_2) = \langle x, 1r_1, r_2\rangle$ where $|r_1| = g(|x|) - 1$ and $|r_2| = y_n$. Note that $f_n$ is a $\frac{1}{2n^2}$-heuristic reduction (because the encodings of triplets form a $\frac{1}{n^2}$ fraction of the set of all strings and because we fix the first bit in the second part of triplet), then by Lemma 5 $L \in \mathrm{Heur}_{1-\frac{1}{n^b}}\mathbf{Size}(((y_n + g(n) + 2\log(n) + n)^k + (n + g(n) + y_n + 2\log(n) + 1))\mathrm{poly}(n))$. Since $L$ is $b$-self-correctible, by Lemma 2 we have $L \in \mathbf{Size}((n + y_n + g(n) + \log(n) + 1)^k \mathrm{poly}(n))$, which contradicts the definition of $s(n)$ (note that since $L \not\subseteq \mathbf{P/poly}$, $y_n$ is infinitely often superpolynomial). □

## Further directions

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

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

Switching to $\mathbf{AM}(= \mathbf{BP} \cdot \mathbf{NP})$ and decreasing the number of random bits in the protocol would derandomize Theorem 1 down to heuristic **NP** and lead consequently to the lower bound $\mathbf{NP} \not\subseteq \mathbf{Size}(n^k)$ for classical computations.

## 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.

## References

[Ajt83] Miklos Ajtai. $\Sigma_1^1$-formulae on finite structures. *Annals of Pure and Applied Logic*, 24:1–48, 1983.

[BFT98] Harry Buhrman, Lance Fortnow, and Thomas Thierauf. Nonrelativizing separations. In *IEEE Conference on Computational Complexity*, pages 8–12. IEEE Computer Society, 1998.

[Blu83] Norbert Blum. A boolean function requiring 3n network size. *Theoretical Computer Science*, 28(3):337 – 345, 1983.

[BT06] Andrej Bogdanov and Luca Trevisan. Average-case complexity. In *in Foundations and Trends in Theoretical Computer Science Volume 2, Issue 1*, 2006.

[Cai01] Jin-Yi Cai. $s_p^2 \subseteq \mathbf{ZPP}^{\mathbf{NP}}$. *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science*, pages 620–629, 2001.

[Hås86] Johan Håstad. Almost optimal lower bounds for small depth circuits. In *ACM STOC*, pages 6–20, 1986.

[Its09] Dmitry Itsykson. Structural complexity of AvgBPP. In *Proceedings of the Fourth International Computer Science Symposium in Russia on Computer Science - Theory and Applications*, CSR '09, pages 155–166, Berlin, Heidelberg, 2009. Springer-Verlag.

[Kan82] Ravi Kannan. Circuit-size lower bounds and non-reducibility to sparse sets. *Information and Control*, 55(1):40–56, 1982.

[Per07] Konstantin Pervyshev. On heuristic time hierarchies. *IEEE Conference of Computational Complexity*, pages 347–358, 2007.

[Raz85] Alexander Razborov. Lower bounds for the monotone complexity of some boolean functions. *Doklady Akademii Nauk SSSR*, 281(4):798–801, 1985.

[San07] Rahul Santhanam. Circuit lower bounds for Merlin-Arthur classes. In *ACM STOC*, pages 275–283, 2007.

[Sha90] Adi Shamir. IP=PSPACE. In *FOCS*, pages 11–15, 1990.

[TV02] Luca Trevisan and Salil Vadhan. Pseudorandomness and average-case complexity via uniform reductions. In *Proceedings of the 17th Annual IEEE Conference on Computational Complexity*, pages 129–138, 2002.