Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 31 (2017)

Quadratic Simulations of Merlin–Arthur Games

Thomas Watson*

April 20, 2020

Abstract

The known proofs of $\mathsf{MA} \subseteq \mathsf{PP}$ incur a quadratic overhead in the running time. We prove that this quadratic overhead is necessary for black-box simulations; in particular, we obtain an oracle relative to which $\mathsf{MA\text{-}TIME}(t) \not\subseteq \mathsf{P\text{-}TIME}(o(t^2))$. We also show that 2-sided-error Merlin–Arthur games can be simulated by 1-sided-error Arthur–Merlin games with quadratic overhead. We also present a simple, query complexity based proof (provided by Mika Göös) that there is an oracle relative to which $\mathsf{MA} \not\subseteq \mathsf{NP}^{\mathsf{BPP}}$ (which was previously known to hold by a proof using generics).

1 Introduction

There are several complexity class inclusions for which all the known proofs consist of "black-box" simulations incurring at least a quadratic overhead in the running time. There have also been lower bounds showing that for some of these inclusions, the quadratic overhead is necessary for black-box simulations (which also yields corresponding oracle separations). We begin by giving an overview of this topic. For convenience we abbreviate "quadratic-overhead black-box simulation" as "quadratic simulation". (Some relevant complexity class definitions can be found in Appendix A.)

- $\mathsf{BPP} \subseteq \Sigma_2\mathsf{P}$ [Sip83, Lau83] holds by quadratic simulations, and Viola [Vio09] proved that the quadratic overhead is necessary. Some known strengthenings of this inclusion include $\mathsf{S}_2\cdot\mathsf{BPP} \subseteq \mathsf{S}_2\mathsf{P}$ [RS98] and the facts that 2-sided-error Merlin–Arthur and Arthur–Merlin games are equivalent to their 1-sided-error counterparts: $\mathsf{MA}_2 \subseteq \mathsf{MA}_1$ and $\mathsf{AM}_2 \subseteq \mathsf{AM}_1$. Of course, the lower bound of [Vio09] also applies to these strengthenings.
- Arthur–Merlin games can simulate Merlin–Arthur games ($\mathsf{MA}_1 \subseteq \mathsf{AM}_1$ and $\mathsf{MA}_2 \subseteq \mathsf{AM}_2$) quadratically [Bab85]. Diehl [Die07] proved that the quadratic overhead is necessary, even for $\mathsf{MA}_1 \subseteq \mathsf{AM}_2$. (As a side result, we complement this by giving a quadratic simulation even for $\mathsf{MA}_2 \subseteq \mathsf{AM}_1$.)
- $\mathsf{MA}_2 \subseteq \mathsf{PP}$ [Ver92] holds by quadratic simulations. As our main result, we prove that the quadratic overhead is necessary (which was stated as an open problem in [Die07]), even for the weaker inclusion $\mathsf{N}\cdot\mathsf{coRP} \subseteq \mathsf{PP}$.[1] A strengthening of the latter inclusion is the quadratic simulation for $\mathsf{P}\cdot\mathsf{BQP} \subseteq \mathsf{PP}$ [FR99].

*Department of Computer Science, University of Memphis. Supported by NSF grant CCF-1657377.

[1] We mention that in the world of communication complexity, a nearly quadratic separation between $\mathsf{N}\cdot\mathsf{coRP}$-type complexity and PP-type complexity is witnessed by the inner product mod 2 function—see [AW09] for the $\mathsf{N}\cdot\mathsf{coRP}$ upper bound and [KN97, §3.5–3.6 and references therein] for the PP lower bound. However, this is not directly relevant to our results since the upper bound is really specific to communication complexity.

ISSN 1433-8092

- PP is closed under intersection by quadratic simulations [BRS95] (for all $L_1, L_2 \in \mathsf{P\text{-}TIME}(n)$ we have $L_1 \cap L_2 \in \mathsf{P\text{-}TIME}(n^2)$). Sherstov [She13] proved that the quadratic overhead is necessary.

1.1 Statement of result

A randomized decision tree with q queries and r random bits consists of a uniform distribution over a multiset of 2^r deterministic decision trees that each make at most q queries on every computation path. A randomized decision tree computes a boolean function with unbounded error if on each input, the output is correct with probability $> 1/2$.

Consider the partial function $F_{\mathsf{N\cdot coR}}$ that takes a $2^n \times 2^n$ boolean matrix with the promise that each row has either all 1's or at most half 1's, and evaluates to 1 if there exists an all-1 row, and to 0 otherwise. Note that $F_{\mathsf{N\cdot coR}}$ is computable by a randomized unbounded-error decision tree with $O(n)$ queries and $O(n^2)$ random bits;[2] this is what underlies the standard proof that $\mathsf{MA_1\text{-}TIME}(n) \subseteq \mathsf{P\text{-}TIME}(n^2)$.

Theorem 1. *Every randomized unbounded-error decision tree for $F_{\mathsf{N\cdot coR}}$ uses either $\Omega(n)$ queries or $2^{\Omega(n)}$ random bits.*

For our interpretation about the necessity of a quadratic overhead (see the corollaries below), it suffices to have $\Omega(n^2)$ random bits (rather than $2^{\Omega(n)}$) at the end of the theorem statement. We conjectured that an $\Omega(n)$ query lower bound holds regardless of the number of random bits, and based on a suggestion from an anonymous reviewer, we can confirm this conjecture by combining our argument with some machinery due to Sherstov [She18]. We explain how to do this after presenting our self-contained proof of Theorem 1 in Section 2.

Corollary 1. *There is an oracle relative to which $\mathsf{N\cdot coR\text{-}TIME}(n) \not\subseteq \mathsf{P\text{-}TIME}(o(n^2))$.*

Corollary 1 holds in the standard model of relativization where the oracle tape is erased after each query. This forces each query to cost linear time, which makes sense in our context since a query is intended to correspond to running a simulation of the deterministic algorithm underlying an $\mathsf{N\cdot coR\text{-}TIME}(n)$ algorithm. Corollary 1 follows in a completely routine way from Theorem 1 (see [Vio09, Die07] for examples of how such diagonalization arguments go).

Our result can also be interpreted in terms of what we call "black-box proofs of $\mathsf{N\cdot coR\text{-}TIME}(n) \subseteq \mathsf{P\text{-}TIME}(t)$". Such a proof consists of a uniform randomized algorithm that takes 1^n as input, computes $F_{\mathsf{N\cdot coR}}$ with unbounded error on an instance it has oracle access to, and runs in time $O(t(n))$ where each oracle query is charged time n. All known proofs of that inclusion are indeed black-box.

Corollary 2. *There is no black-box proof of $\mathsf{N\cdot coR\text{-}TIME}(n) \subseteq \mathsf{P\text{-}TIME}(o(n^2))$.*

[2] First consider picking uniformly at random a row and a sequence of $n+1$ bits from that row, and accepting iff all those bits are 1. This accepts 1-inputs with probability $\geq 1/2^n$ and 0-inputs with probability $\leq 1/2^{n+1}$. Now modify this by using $n+3$ more uniformly random bits, interpreted as the binary representation of an integer $k < 2^{n+3}$: if $k < 2^{n+2} - 3$ then accept; if $k \geq 2^{n+2}$ then run the algorithm from the previous sentence; otherwise reject. The modified algorithm accepts 1-inputs with probability $\geq 1/2 - 3/2^{n+3} + 1/2^{n+1} > 1/2$ and 0-inputs with probability $\leq 1/2 - 3/2^{n+3} + 1/2^{n+2} < 1/2$.

Corollary 2 follows immediately from Corollary 1 since black-box proofs relativize. Corollary 2 also follows directly from Theorem 1 since such a black-box proof is just a uniform, time-efficient implementation of a randomized unbounded-error decision tree for $F_{\mathsf{N}\cdot\mathsf{coR}}$ that uses $o(n)$ queries and $o(n^2)$ random bits.

For convenience, we have focused on time n vs. n^2, but our lower bound also works for any time-constructible $t(n)$ vs. $t(n)^2$.

1.2 Relevance to time-space lower bounds

There is a line of research on time-space lower bounds for problems related to satisfiability [van06]. One of the motivations for Viola [Vio09] to initiate the study of quadratic simulation lower bounds was that they can provide barriers to improving such time-space lower bounds. It is known that for every constant $\epsilon > 0$,

(i) SAT (which is NP-complete) cannot be solved by a deterministic algorithm running in time $n^{2\cos(\pi/7)-\epsilon} \approx n^{1.8019}$ and space $n^{o(1)}$ [Wil08];

(ii) Σ_2SAT (which is $\Sigma_2\mathsf{P}$-complete) cannot be solved by a bounded-error randomized algorithm running in time $n^{2-\epsilon}$ and space $n^{o(1)}$ [Dv06];

(iii) MajMajSAT (which is $\mathsf{P}\cdot\mathsf{PP}$-complete) cannot be solved by a bounded-error quantum algorithm running in time $n^{1+o(1)}$ and space $n^{1-\epsilon}$ [vW12, AKR+01].

It is open to prove a nontrivial randomized time-space lower bound for SAT rather than Σ_2SAT (the first rather than the second level of the polynomial hierarchy). A natural approach to prove this (following [Dv06]) would involve "swapping Arthur and Merlin" (i.e., using $\mathsf{MA}_2 \subseteq \mathsf{AM}_2$); however, the quadratic overhead is too inefficient to yield any nontrivial lower bound. Indeed, one of the motivations for the result of [Die07] is that it implies this approach cannot be made to work via a subquadratic black-box simulation.

Similarly, it is open to prove a nontrivial quantum time-space lower bound for MajSAT rather than MajMajSAT (the first rather than the second level of the counting hierarchy). A natural approach to prove this (following [vW12]) would involve "absorbing quantumness into a majority quantifier" (i.e., using $\mathsf{P}\cdot\mathsf{BQP} \subseteq \mathsf{PP}$ [FR99]); however, the quadratic overhead is too inefficient to yield any nontrivial lower bound. Our result implies this approach cannot be made to work via a subquadratic black-box simulation (since $\mathsf{N}\cdot\mathsf{coR}\text{-}\mathsf{TIME}(n) \subseteq \mathsf{P}\cdot\mathsf{BQ}\text{-}\mathsf{TIME}(n)$).

2 Proof of Theorem 1

Suppose for contradiction that $F_{\mathsf{N}\cdot\mathsf{coR}}$ has a randomized unbounded-error decision tree using $\leq n/6$ queries and $\leq 2^{n/4}$ uniformly random bits. Such a decision tree can be expressed as a polynomial threshold function (PTF) with integer coefficients, having degree $\leq n/6$ and weight $\leq 2^{2^{n/3}}$ (the weight is the sum of the absolute values of the coefficients).[3] We use a two-step argument: first, we show that a particular approach for designing such a PTF fails; second, we essentially show that if

[3] Specifically, the polynomial is the sum over all deterministic decision trees in the support, and over all that tree's accepting computation paths, of the conjunction expressing that path. The bound on the weight has a factor of $2^{2^{n/4}}$ since there are $\leq 2^{n/4}$ random bits; rounding this up to $2^{2^{n/3}}$ loosely accounts for the deterministic decision trees' contributions to the weight.

that approach fails then *every* approach fails (by using an adaptation of Vereshchagin's machinery from [Ver95]).

If there were a univariate polynomial p of degree $\leq n/6$ such that $p(2^n) > 2^n$ and $p(i) \in [0,1]$ for all $i \in \{0,1,2,\ldots,2^{n-1}\}$, then we could get a PTF of degree $\leq n/6$ for $F_{\mathsf{N}\cdot\mathsf{coR}}$ by taking the sum over all rows of p applied to the sum of the bits in that row, and using 2^n as the threshold. (Moreover, if the coefficients of p were all integer multiples of some $a > 0$ and p had weight $\leq a2^{2^{n/4}}$, then we could use p/a to get a PTF having weight $\leq 2^{2^{n/3}}$.) However, this approach cannot work:

Claim 1. *There is no univariate polynomial p of degree $\leq n/6$ such that $p(2^n) > 2^{n/2}$ and $p(i) \in [0,1]$ for all $i \in \{0,1,2,\ldots,2^{n-1}\}$.*

Proof. Let us modify p by transforming the input interval $[0,2^{n-1}]$ to $[-1,1]$ and shifting the graph down by $1/2$, i.e., define the polynomial $q(x) \coloneqq p\big((x+1)2^{n-2}\big) - 1/2$. Then we have $q(3) > 2^{n/2} - 1/2$ and $q(-1+i/2^{n-2}) \in [-1/2, 1/2]$ for all $i \in \{0,1,2,\ldots,2^{n-1}\}$. We claim the latter property implies that for all $x \in [-1,1]$ we have

$$|q(x)| \;\leq\; (1/2)/\big(1 - O(\deg(q)^2/2^n)\big) \;\leq\; 1. \tag{1}$$

The second inequality holds since $\deg(q) = \deg(p) \leq n/6 \leq o(2^{n/2})$. The first inequality is by a standard result that has been widely used in the literature (starting with [NS94]) and is generally attributed to [EZ64, RC66]. Specifically, taking maxima over $x \in [-1,1]$, we have

$$2^{n-1}(\max|q(x)| - 1/2) \;\leq\; \max|q'(x)| \;\leq\; \deg(q)^2 \cdot \max|q(x)| \tag{2}$$

where the first inequality of (2) is by the mean value theorem and the second inequality of (2) is by Markov's inequality in approximation theory. Rearranging gives the first inequality of (1).

In summary, $q(3) > 2^{n/2} - 1/2$, $|q(x)| \leq 1$ for all $x \in [-1,1]$, and $\deg(q) \leq n/6$. To show that this is impossible, we appeal to a classic result stating that Chebyshev polynomials are extremal in the following sense (see [Riv81, Theorem 1.10] or [Car, Theorem 4.12] for a proof): If T_d is the degree-d Chebyshev polynomial of the first kind (defined by the recurrence $T_0(x) \coloneqq 1$, $T_1(x) \coloneqq x$, and $T_{d+1}(x) \coloneqq 2xT_d(x) - T_{d-1}(x)$ for $d \geq 1$) and q is any degree-d polynomial such that $|q(x)| \leq 1$ for all $x \in [-1,1]$, then for all $x \geq 1$ we have $|q(x)| \leq T_d(x)$. To get a contradiction, note that the recurrence trivially implies that $T_d(3) \leq 6^d$, and thus $q(3) \leq 6^d \leq 2^{n/2} - 1/2$ for $d \leq n/6$. ☐

Now we begin the "bootstrapping," using Claim 1 to show that not only does the most natural approach to designing a PTF fail, but *every* approach fails.

Claim 2. *There exist distributions D_0 and D_1 over $\{0,1,2,\ldots,2^{n-1}\} \cup \{2^n\}$ such that $\mathbb{P}_{D_0}[2^n] = 0$, $\mathbb{P}_{D_1}[2^n] = 2^{-n/2}$, and $\mathbb{E}_{i\sim D_0}[i^k] = \mathbb{E}_{i\sim D_1}[i^k]$ for all $k \in \{0,1,2,\ldots,n/6\}$.*

Proof. The claim is equivalent to the feasibility of the following system with variables v_i and w_i for $i \in \{0,1,2,\ldots,2^{n-1}\}$ (representing $\mathbb{P}_{D_0}[i]$ and $\mathbb{P}_{D_1}[i]$ respectively), where we define $\delta \coloneqq 2^{-n/2}$.

$$\begin{aligned}
\textstyle\sum_i v_i &= 1 \\
\textstyle\sum_i w_i &= 1 - \delta \\
\textstyle\sum_i v_i \cdot i^k - \sum_i w_i \cdot i^k &= \delta \cdot (2^n)^k \qquad \text{for all } k \in \{0,1,2,\ldots,n/6\} \\
v_i,\; w_i &\geq 0 \qquad \text{for all } i \in \{0,1,2,\ldots,2^{n-1}\}
\end{aligned}$$

By Farkas's Lemma, this is equivalent to the infeasibility of the following system with variables x, y, and z_k for $k \in \{0, 1, 2, \ldots, n/6\}$.

$$\begin{aligned} x + \textstyle\sum_k z_k \cdot i^k &\geq 0 \qquad \text{for all } i \in \{0, 1, 2, \ldots, 2^{n-1}\} \\ y - \textstyle\sum_k z_k \cdot i^k &\geq 0 \qquad \text{for all } i \in \{0, 1, 2, \ldots, 2^{n-1}\} \\ x + y \cdot (1 - \delta) + \textstyle\sum_k z_k \cdot \delta \cdot (2^n)^k &< 0 \end{aligned}$$

Defining the polynomial $Z(i) \coloneqq \sum_k z_k \cdot i^k$, this system can be rewritten as follows.

$$Z(i) \in [-x, y] \qquad \text{for all } i \in \{0, 1, 2, \ldots, 2^{n-1}\} \tag{3}$$

$$x + y \cdot (1 - \delta) + \delta \cdot Z(2^n) < 0 \tag{4}$$

Suppose for contradiction this system is feasible; in particular $y \geq -x$. We cannot have $y = -x$ since then by (3), Z would either be the constant $y = -x$, thus violating (4), or have degree $> 2^{n-1} > n/6$. Thus we may assume $x+y > 0$. If we define the polynomial $Z^*(i) \coloneqq (y - Z(i))/(x+y)$ then $Z^*(i) \in [0, 1]$ for all $i \in \{0, 1, 2, \ldots, 2^{n-1}\}$ by (3), and $Z^*(2^n) > 1/\delta = 2^{n/2}$ by (4); yet $\deg(Z^*) = \deg(Z) \leq n/6$, contradicting Claim 1. □

For $b \in \{0, 1\}$, define μ_b as the distribution over $2^n \times 2^n$ boolean matrices M obtained by, for each row independently, sampling $i \sim D_b$ and then taking a uniformly random length-2^n bit string of Hamming weight i. Let "$P(M) > t$" be the purported PTF for $F_{\mathsf{N \cdot coR}}$ (where t is an integer). The following two claims provide a contradiction.

Claim 3. $\mathbb{E}_{\mu_1}[P(M)] > \mathbb{E}_{\mu_0}[P(M)]$.

Claim 4. $\mathbb{E}_{\mu_1}[P(M)] = \mathbb{E}_{\mu_0}[P(M)]$.

Proof of Claim 3. Let us abbreviate $F_{\mathsf{N \cdot coR}}$ as F. Observe that $\mathbb{P}_{\mu_0}\left[F^{-1}(0)\right] = 1$ and $\mathbb{P}_{\mu_1}\left[F^{-1}(1)\right] = 1 - (1 - 2^{-n/2})^{2^n} \geq 1 - e^{-2^{n/2}}$. Also, notice that $|P(M)| \leq \text{weight}(P) \leq 2^{2^{n/3}}$ for all M; in particular, $t < 2^{2^{n/3}}$. Thus,

$$\begin{aligned} \mathbb{E}_{\mu_1}[P(M)] &= \mathbb{E}_{\mu_1}\left[P(M) \,\middle|\, F^{-1}(1)\right] \cdot \mathbb{P}_{\mu_1}\left[F^{-1}(1)\right] + \mathbb{E}_{\mu_1}\left[P(M) \,\middle|\, F^{-1}(0)\right] \cdot \mathbb{P}_{\mu_1}\left[F^{-1}(0)\right] \\ &\geq (t+1) \cdot \left(1 - e^{-2^{n/2}}\right) - 2^{2^{n/3}} \cdot e^{-2^{n/2}} \\ &> t \\ &\geq \mathbb{E}_{\mu_0}[P(M)]. \end{aligned}$$

□

Proof of Claim 4. Define U_i to be the uniform distribution over length-2^n bit strings of Hamming weight i. For any $C \subseteq [2^n]$, we have $\mathbb{P}_{u \sim U_i}\left[u_C \text{ is all 1's}\right] = \frac{i(i-1)\cdots(i-|C|+1)}{2^n(2^n-1)\cdots(2^n-|C|+1)}$ (most easily seen by imagining that u is fixed and C is random); this is a polynomial of degree $|C|$ in i, which we write as $Q^{|C|}(i) \coloneqq \sum_{k=0}^{|C|} Q_k^{|C|} \cdot i^k$. We also write $P(M) \coloneqq \sum_S P_S \prod_{(r,c) \in S} M_{r,c}$ where the sum ranges over $S \subseteq [2^n] \times [2^n]$ with $|S| \leq n/6$. For a row index $r \in [2^n]$, let $S_r \coloneqq \left\{c \in [2^n] : (r, c) \in S\right\}$. For each $b \in \{0, 1\}$ we have

$$\begin{aligned} \mathbb{E}_{\mu_b}[P(M)] &= \textstyle\sum_S P_S \, \mathbb{P}_{\mu_b}\left[M_S \text{ is all 1's}\right] \\ &= \textstyle\sum_S P_S \prod_r \mathbb{E}_{i \sim D_b} \mathbb{P}_{u \sim U_i}\left[u_{S_r} \text{ is all 1's}\right] \end{aligned}$$

$$= \sum_S P_S \prod_r \mathbb{E}_{i\sim D_b}\left[Q^{|S_r|}(i)\right]$$
$$= \sum_S P_S \prod_r \sum_k Q_k^{|S_r|} \mathbb{E}_{i\sim D_b}[i^k].$$

By Claim 2, this value does not depend on b (since $k \leq |S_r| \leq |S| \leq n/6$ always holds). □

This concludes the proof of Theorem 1. We now explain how to strengthen the result to show that every randomized unbounded-error decision tree for $F_{\mathsf{N\cdot coR}}$ uses $\Omega(n)$ queries, no matter how many random bits (although this is unnecessary for Corollary 1 and Corollary 2). We use the following definition and technical lemma due to Sherstov.

Definition 1. *The* one-sided ϵ-approximate degree *of a partial function* $f\colon \{0,1\}^L \to \{0,1\}$*, denoted* $\deg_\epsilon^+(f)$*, is the least degree of a real polynomial* P *such that* $|P(x)| \leq \epsilon$ *for* $x \in f^{-1}(0)$ *and* $P(x) \geq 1-\epsilon$ *for* $x \in f^{-1}(1)$*.*

The *block composition* of $g\colon \{0,1\}^N \to \{0,1\}$ and $f\colon \{0,1\}^L \to \{0,1\}$ is the (partial) function $g \circ f^N\colon (\{0,1\}^L)^N \to \{0,1\}$ defined by $(g \circ f^N)(x_1,\ldots,x_N) = g(f(x_1),\ldots,f(x_N))$. The following, which concerns the $\textsc{Or}_N$ function on N bits, is a special case of [She18, Corollary 6.10 (or Corollary 6.8 in the ECCC version)].

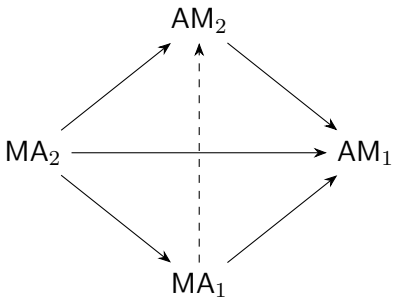
Lemma 1. *For all* f *and* $\epsilon > 0$*, every randomized unbounded-error decision tree for* $\textsc{Or}_N \circ f^N$ *uses* $\Omega(\min(\deg_\epsilon^+(f), N\sqrt{\epsilon}))$ *queries.*

Defining $\textsc{GapAnd}_L$ to be the partial function that evaluates to 1 if all input bits are 1, and evaluates to 0 if at most half of the input bits are 1, we have $F_{\mathsf{N\cdot coR}} = \textsc{Or}_{2^n} \circ \textsc{GapAnd}_{2^n}^{2^n}$. Combining Claim 1 with the classic symmetrization technique [MP87, Bd02] shows that $\deg_\epsilon^+(\textsc{GapAnd}_{2^n}) \geq \Omega(n)$ for $\epsilon = 1/2^{n/2}$. Combining this with Lemma 1 implies that every randomized unbounded-error decision tree for $\textsc{Or}_{2^n} \circ \textsc{GapAnd}_{2^n}^{2^n}$ uses $\Omega(\min(n, 2^{3n/4})) = \Omega(n)$ queries.

3 Quadratic Simulation for $\mathsf{MA}_2 \subseteq \mathsf{AM}_1$

Theorem 2. $\mathsf{MA}_2\text{-}\mathsf{TIME}(n) \subseteq \mathsf{AM}_1\text{-}\mathsf{TIME}(n^2)$.

For historical context, the four diagonal arrows in the figure represent known simulations with quadratic overhead: $\mathsf{MA}_2\text{-}\mathsf{TIME}(n) \subseteq \mathsf{MA}_1\text{-}\mathsf{TIME}(n^2 \operatorname{polylog} n)$ and $\mathsf{AM}_2\text{-}\mathsf{TIME}(n) \subseteq \mathsf{AM}_1\text{-}\mathsf{TIME}(n^2 \operatorname{polylog} n)$ follow by the "covering by shifts" argument of [Lau83], while $\mathsf{MA}_2\text{-}\mathsf{TIME}(n) \subseteq \mathsf{AM}_2\text{-}\mathsf{TIME}(n^2)$ and $\mathsf{MA}_1\text{-}\mathsf{TIME}(n) \subseteq \mathsf{AM}_1\text{-}\mathsf{TIME}(n^2)$ follow by amplification and swapping the quantifiers, as shown in [Bab85]. The horizontal arrow represents Theorem 2. The dashed vertical arrow represents the lower bound of [Die07] showing that black-box or relativizing techniques cannot even yield $\mathsf{MA}_1\text{-}\mathsf{TIME}(n) \subseteq \mathsf{AM}_2\text{-}\mathsf{TIME}(o(n^2))$.



Of course, a 4th-power simulation for $\mathsf{MA}_2 \subseteq \mathsf{AM}_1$ follows from the previous results, by carrying out the two steps (swapping the quantifiers and making the error 1-sided) in either order. To prove Theorem 2 we need a single quadratic simulation that handles both steps at the same time. Our

proof ends up resembling the proof of $\mathsf{S}_2\cdot\mathsf{BPP} \subseteq \mathsf{S}_2\mathsf{P}$ in [RS98], but (similarly to [Dv06]) we start by doing randomness-efficient amplification with very explicit expanders, and we also set parameters differently (in particular, using constant numbers of shifts).

We first state the randomness-efficient amplification result we use. An *expander* is defined as an infinite family of constant-degree undirected graphs, each of whose normalized adjacency matrix has at least a positive constant gap between the first and second eigenvalues.

Lemma 2. *Any $O(n)$-time constant-error randomized algorithm can be amplified to have error probability $< 2^{-n}$ while using $O(n)$ random bits and running in time $O(n^2)$. This follows by combining the following two ingredients:*

- *[CW89, IZ89] A constant-error randomized algorithm can be amplified to have error probability $< \varepsilon$ by taking an $O(\log(1/\varepsilon))$-step random walk on an expander whose vertices correspond to outcomes of the original algorithm's randomness, running the original algorithm using each of the outcomes on the walk, and outputting the majority vote.*
- *[GG81] There is an expander whose vertices are bit strings of length $O(n)$, where each vertex's neighborhood can be computed in time $O(n)$.*

Specifically, the first statement in Lemma 2 follows by taking an $O(n)$-step random walk on the expander, using $O(n)$ random bits to specify the initial vertex and $O(n)$ random bits to specify the edge labels on the walk, and for each step using $O(n)$ time to compute the vertex from the previous one and $O(n)$ time to run the original algorithm again.

Proof of Theorem 2. By the randomness-efficient amplification of Lemma 2, we may assume that Arthur has error probability $< 2^{-n}$ while using $O(n)$ random bits and running in time $O(n^2)$. That is, for $L \in \mathsf{MA}_2\text{-}\mathsf{TIME}(n)$ there is a deterministic $O(n^2)$-time algorithm M and a constant c such that if $x \in L$ then $\exists w \in \{0,1\}^{cn}\ \mathbb{P}_{r\in\{0,1\}^{cn}}[M(x,w,r) \text{ accepts}] > 1-2^{-n}$, and if $x \notin L$ then $\forall w \in \{0,1\}^{cn}\ \mathbb{P}_{r\in\{0,1\}^{cn}}[M(x,w,r) \text{ accepts}] < 2^{-n}$. Consider the $O(n^2)$-time algorithm M' that, letting $a \coloneqq c^2+c+1$ and $b \coloneqq c$, interprets its input as $x \in \{0,1\}^n$, $r' \coloneqq r_1 \cdots r_a \in (\{0,1\}^{cn})^a$, and $w' \coloneqq ws_1 \cdots s_b \in \{0,1\}^{cn} \times (\{0,1\}^{cn})^b$, and accepts iff $\forall i \in [a]\ \exists j \in [b]\ M(x,w,r_i \oplus s_j)$ accepts. We claim that M' witnesses $L \in \mathsf{AM}_1\text{-}\mathsf{TIME}(n^2)$. First suppose $x \in L$, and fix $w \in \{0,1\}^{cn}$ such that $\mathbb{P}_r[M(x,w,r) \text{ accepts}] > 1-2^{-n}$. If we pick $s_1 \cdots s_b \in (\{0,1\}^{cn})^b$ uniformly at random, then for each $r \in \{0,1\}^{cn}$ we have $\mathbb{P}_{s_1\cdots s_b}\left[\neg\exists j\ M(x,w,r\oplus s_j) \text{ accepts}\right] < (2^{-n})^b = 2^{-cn}$. Hence by a union bound, there exists $s_1 \cdots s_b$ such that for all r there exists a j such that $M(x,w,r \oplus s_j)$ accepts. Letting $w' \coloneqq ws_1 \cdots s_b$, we have $\exists w'\ \forall r'\ M'(x,r',w')$ accepts, and thus $\mathbb{P}_{r'}\left[\exists w'\ M'(x,r',w') \text{ accepts}\right] = 1$. Now suppose $x \notin L$. If we pick $r' \in (\{0,1\}^{cn})^a$ uniformly at random, then for each $w' \coloneqq ws_1 \cdots s_b$ we have $\mathbb{P}_{r'}\left[\forall i\ \exists j\ M(x,w,r_i \oplus s_j) \text{ accepts}\right] < (b2^{-n})^a \leq \frac{1}{2}\cdot 2^{-(cn+bcn)}$. By a union bound, $\mathbb{P}_{r'}\left[\exists w'\ M'(x,r',w') \text{ accepts}\right] \leq 1/2$. ∎

4 Relativized $\mathsf{MA} \not\subseteq \mathsf{NP}^{\mathsf{BPP}}$

The distinction between MA_1 and $\mathsf{N}\cdot\mathsf{coRP}$ is that when Merlin sends a "wrong" witness for a 1-input, MA_1 allows Arthur to accept with arbitrary probability, whereas $\mathsf{N}\cdot\mathsf{coRP}$ requires Arthur to accept with a "legal" probability (in $[0,1/2] \cup \{1\}$). The distinction between MA_2 and $\mathsf{N}\cdot\mathsf{BPP}$ is similar but where the legal probabilities are $[0,1/3] \cup [2/3,1]$. Since the relativizing polynomial-time class equalities $\mathsf{MA} \coloneqq \mathsf{MA}_2 = \mathsf{MA}_1$ and $\mathsf{NP}^{\mathsf{BPP}} = \mathsf{N}\cdot\mathsf{BPP} = \mathsf{N}\cdot\mathsf{coRP}$ hold, the following theorem shows that the distinction is significant.

Theorem 3. *There is an oracle relative to which* $\mathsf{MA} \not\subseteq \mathsf{NP}^{\mathsf{BPP}}$.

Theorem 3 was shown in [FFKL03] using the machinery of generics. In contrast, most oracle separations of pairs of ordinary complexity classes are known to hold directly via separations of the corresponding query complexity (decision tree) models. When we asked Mika Göös whether a query complexity style argument could be used to prove Theorem 3, he promptly manufactured such an argument. He declined coauthorship but graciously gave permission to present the argument for the sake of recording it in the literature. Furthermore, this argument even yields an oracle relative to which $\mathsf{MA} \not\subseteq \mathsf{NP}^{\mathsf{BQP}}$ (by using a quantum rather than randomized query lower bound for the OR function in the appropriate places), which appears to be a new result.

We define a q-query $\mathsf{N{\cdot}BPP}$ decision tree for a partial function $F\colon \{0,1\}^N \to \{0,1\}$ to be a set of probability distributions over depth-q deterministic decision trees, such that for every 0-input, each distribution in the set accepts with probability $\leq 1/3$, and for every 1-input, each distribution in the set accepts with probability in $[0,1/3] \cup [2/3,1]$ and at least one of them accepts with probability $\geq 2/3$.[4]

Consider the partial function F_{MA_1} that takes a $2^n \times 2^n$ boolean matrix and evaluates to 1 if there exists an all-1 row, and to 0 if each row has at most half 1's. Theorem 3 is a corollary of the following result, by the standard connection between decision tree lower bounds and oracle separations [Ver99].

Lemma 3. *Every* $\mathsf{N{\cdot}BPP}$ *decision tree for* F_{MA_1} *uses* $\Omega(2^n)$ *queries.*

Proof. Let us abbreviate F_{MA_1} as F. Suppose for contradiction there exists an $\mathsf{N{\cdot}BPP}$ decision tree for F using at most 2^{n-4} queries. Define $M^{(0)}$ to be the $2^n \times 2^n$ matrix that has all 1's in its first row and 0's everywhere else. Since $F(M^{(0)}) = 1$, there is a distribution D (which we fix henceforth) in the set of the $\mathsf{N{\cdot}BPP}$ decision tree, that accepts $M^{(0)}$ with probability $\geq 2/3$.

We claim that there exists a sequence of $2^n \times 2^n$ matrices $M^{(0)}, M^{(1)}, \ldots, M^{(2^{n-1})}$ such that for each $i = 1, \ldots, 2^{n-1}$, $M^{(i)}$ has $2^n - i$ 1's in its first row and 0's everywhere else, and the probability D accepts $M^{(i)}$ is within $1/8$ of the probability D accepts $M^{(i-1)}$. Inductively assuming $M^{(i-1)}$ has been constructed, there must be a 1-entry that gets queried with probability $\leq 2^{n-4}/(2^n - (i-1)) \leq 1/8$ under D (since the sum over the $2^n - (i-1)$ many 1 entries of the probability it gets queried is at most the worst-case number of queries, which is $\leq 2^{n-4}$), so we can obtain $M^{(i)}$ by flipping this entry to a 0. The claim is proved.

Since $F(M^{(2^{n-1})}) = 0$, D accepts $M^{(2^{n-1})}$ with probability $\leq 1/3$. Hence there exists an i^* such that D accepts $M^{(i^*)}$ with probability within $1/16$ of $1/2$. This is an illegal probability, but we do not yet have a contradiction, since $M^{(i^*)}$ is not in the domain of F. Now there must be a row of $M^{(i^*)}$ such that the probability (under D) that a bit in that row gets queried is $\leq 2^{n-4}/2^n = 1/16$. Flipping all the 0's to 1's in that row results in a matrix M that D accepts with (illegal) probability within $1/16 + 1/16$ of $1/2$. This is a contradiction since $F(M) = 1$ and so D is supposed to accept M with probability in $[0,1/3] \cup [2/3,1]$. □

[4] It would also be natural to charge the log of the size of the set—i.e., the number of nondeterministic guess bits—to the cost of the decision tree. For Theorem 3 it would suffice to consider this more restricted model, but our lower bound holds even for the more powerful model that does not charge for guess bits. [RTVV99] explores this distinction in the context of MA decision trees.

Acknowledgments

I thank Mika Göös for suggesting the proof of Lemma 3, and anonymous reviewers for helpful comments.

A Definitions

In these definitions, we assume any reasonable model of algorithms that have random access to the input and memory. Probabilities are always over a uniform distribution.

Definition. $L \in \mathsf{MA}_1\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] = 1 \\ x \notin L &\Rightarrow \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \leq 1/2. \end{aligned}$$

Definition. $L \in \mathsf{MA}_2\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \geq 2/3 \\ x \notin L &\Rightarrow \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \leq 1/3. \end{aligned}$$

Definition. $L \in \mathsf{N}\cdot\mathsf{coR}\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] = 1 \text{ and } \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \in [0,1/2] \cup \{1\} \\ x \notin L &\Rightarrow \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \leq 1/2. \end{aligned}$$

Definition. $L \in \mathsf{N}\cdot\mathsf{BP}\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \geq 2/3 \text{ and } \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \in [0,1/3] \cup [2/3,1] \\ x \notin L &\Rightarrow \forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \leq 1/3. \end{aligned}$$

Definition. $L \in \mathsf{AM}_1\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \mathbb{P}_y\big[\exists z\ M(x,y,z) \text{ accepts}\big] = 1 \\ x \notin L &\Rightarrow \mathbb{P}_y\big[\exists z\ M(x,y,z) \text{ accepts}\big] \leq 1/2. \end{aligned}$$

Definition. $L \in \mathsf{AM}_2\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \mathbb{P}_y\big[\exists z\ M(x,y,z) \text{ accepts}\big] \geq 2/3 \\ x \notin L &\Rightarrow \mathbb{P}_y\big[\exists z\ M(x,y,z) \text{ accepts}\big] \leq 1/3. \end{aligned}$$

Definition. $L \in \mathsf{S}_2\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \forall z\ M(x,y,z) \text{ accepts} \\ x \notin L &\Rightarrow \exists z\ \forall y\ M(x,y,z) \text{ rejects.} \end{aligned}$$

Definition. $L \in \mathsf{S}_2\cdot\mathsf{BP}\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z, w *where* $|y| = |z| = |w| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \forall z\ \mathbb{P}_w[M(x,y,z,w) \text{ accepts}] \geq 2/3 \\ x \notin L &\Rightarrow \exists z\ \forall y\ \mathbb{P}_w[M(x,y,z,w) \text{ accepts}] \leq 1/3. \end{aligned}$$

Definition. $L \in \Sigma_2\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \exists y\ \forall z\ M(x,y,z) \text{ accepts} \\ x \notin L &\Rightarrow \forall y\ \exists z\ M(x,y,z) \text{ rejects.} \end{aligned}$$

Definition. $L \in \mathsf{P}\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y *where* $|y| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} x \in L &\Rightarrow \mathbb{P}_y[M(x,y) \text{ accepts}] > 1/2 \\ x \notin L &\Rightarrow \mathbb{P}_y[M(x,y) \text{ accepts}] < 1/2. \end{aligned}$$

Definition. $L \in \mathsf{P}\cdot\mathsf{BQ}\text{-}\mathsf{TIME}(t)$ *iff there exists a quantum algorithm* M *that takes inputs* x, y *where* $|y| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} &\forall y\ \mathbb{P}[M(x,y) \text{ accepts}] \in [0, 1/3] \cup [2/3, 1] \\ x \in L &\Rightarrow \mathbb{P}_y\big[\mathbb{P}[M(x,y) \text{ accepts}] \geq 2/3\big] > 1/2 \\ x \notin L &\Rightarrow \mathbb{P}_y\big[\mathbb{P}[M(x,y) \text{ accepts}] \geq 2/3\big] < 1/2. \end{aligned}$$

Definition. $L \in \mathsf{P}\cdot\mathsf{P}\text{-}\mathsf{TIME}(t)$ *iff there exists a deterministic algorithm* M *that takes inputs* x, y, z *where* $|y| = |z| = O(t(|x|))$*, runs in time* $O(t(|x|))$*, and such that for every* x*,*

$$\begin{aligned} &\forall y\ \mathbb{P}_z[M(x,y,z) \text{ accepts}] \neq 1/2 \\ x \in L &\Rightarrow \mathbb{P}_y\big[\mathbb{P}_z[M(x,y,z) \text{ accepts}] > 1/2\big] > 1/2 \\ x \notin L &\Rightarrow \mathbb{P}_y\big[\mathbb{P}_z[M(x,y,z) \text{ accepts}] > 1/2\big] < 1/2. \end{aligned}$$

MA_1, MA_2, $\mathsf{N}\cdot\mathsf{coRP}$, $\mathsf{N}\cdot\mathsf{BPP}$, AM_1, AM_2, $\mathsf{S}_2\mathsf{P}$, $\mathsf{S}_2\cdot\mathsf{BPP}$, $\Sigma_2\mathsf{P}$, PP, $\mathsf{P}\cdot\mathsf{BQP}$, and $\mathsf{P}\cdot\mathsf{PP}$ are defined as the corresponding polynomial time bounded classes. It is known that $\mathsf{MA}_2 = \mathsf{MA}_1$, so MA refers to this common class. Similarly, it is known that $\mathsf{NP}^{\mathsf{BPP}} = \mathsf{N}\cdot\mathsf{BPP} = \mathsf{N}\cdot\mathsf{coRP}$. Also, $\mathsf{AM}_1\text{-}\mathsf{TIME}$ could be called $\mathsf{coR}\cdot\mathsf{N}\text{-}\mathsf{TIME}$, and $\mathsf{AM}_2\text{-}\mathsf{TIME}$ could be called $\mathsf{BP}\cdot\mathsf{N}\text{-}\mathsf{TIME}$. It is known that $\mathsf{AM}_2 = \mathsf{AM}_1$, so AM refers to this common class.

References

[AKR+01] Eric Allender, Michal Koucký, Detlef Ronneburger, Sambuddha Roy, and V. Vinay. Time-space tradeoffs in the counting hierarchy. In *Proceedings of the 16th Conference on Computational Complexity (CCC)*, pages 295–302. IEEE, 2001. doi:10.1109/CCC.2001.933896.

[AW09] Scott Aaronson and Avi Wigderson. Algebrization: A new barrier in complexity theory. *ACM Transactions on Computation Theory*, 1(1):2:1–2:54, 2009. doi:10.1145/1490270.1490272.

[Bab85] László Babai. Trading group theory for randomness. In *Proceedings of the 17th Symposium on Theory of Computing (STOC)*, pages 421–429. ACM, 1985. doi:10.1145/22145.22192.

[Bd02] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002. doi:10.1016/S0304-3975(01)00144-X.

[BRS95] Richard Beigel, Nick Reingold, and Daniel Spielman. PP is closed under intersection. *Journal of Computer and System Sciences*, 50(2):191–202, 1995. doi:10.1006/jcss.1995.1017.

[Car] Neal Carothers. A short course on approximation theory. Lecture notes. URL: http://personal.bgsu.edu/~carother/Approx.html.

[CW89] Aviad Cohen and Avi Wigderson. Dispersers, deterministic amplification, and weak random sources. In *Proceedings of the 30th Symposium on Foundations of Computer Science (FOCS)*, pages 14–19. IEEE, 1989. doi:10.1109/SFCS.1989.63449.

[Die07] Scott Diehl. Lower bounds for swapping Arthur and Merlin. In *Proceedings of the 11th International Workshop on Randomization and Computation (RANDOM)*, pages 449–463. Springer, 2007. doi:10.1007/978-3-540-74208-1_33.

[Dv06] Scott Diehl and Dieter van Melkebeek. Time-space lower bounds for the polynomial-time hierarchy on randomized machines. *SIAM Journal on Computing*, 36(3):563–594, 2006. doi:10.1137/050642228.

[EZ64] Hartmut Ehlich and Karl Zeller. Schwankung von polynomen zwischen gitterpunkten. *Mathematische Zeitschrift*, 86(1):41–44, 1964.

[FFKL03] Stephen Fenner, Lance Fortnow, Stuart Kurtz, and Lide Li. An oracle builder's toolkit. *Information and Computation*, 182(2):95–136, 2003. doi:10.1016/S0890-5401(03)00018-X.

[FR99] Lance Fortnow and John Rogers. Complexity limitations on quantum computation. *Journal of Computer and System Sciences*, 59(2):240–252, 1999. doi:10.1006/jcss.1999.1651.

[GG81] Ofer Gabber and Zvi Galil. Explicit constructions of linear-sized superconcentrators. *Journal of Computer and System Sciences*, 22(3):407–420, 1981. doi:10.1016/0022-0000(81)90040-4.

[IZ89] Russell Impagliazzo and David Zuckerman. How to recycle random bits. In *Proceedings of the 30th Symposium on Foundations of Computer Science (FOCS)*, pages 248–253. IEEE, 1989. doi:10.1109/SFCS.1989.63486.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[Lau83] Clemens Lautemann. BPP and the polynomial hierarchy. *Information Processing Letters*, 17(4):215–217, 1983. doi:10.1016/0020-0190(83)90044-3.

[MP87] Marvin Minsky and Seymour Papert. *Perceptrons—An Introduction to Computational Geometry*. MIT Press, 1987.

[NS94] Noam Nisan and Mario Szegedy. On the degree of boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994. doi:10.1007/BF01263419.

[RC66] Theodore Rivlin and Elliot Ward Cheney. A comparison of uniform approximations on an interval and a finite subset thereof. *SIAM Journal on Numerical Analysis*, 3(2):311–320, 1966.

[Riv81] Theodore Rivlin. *An Introduction to the Approximation of Functions*. Dover, 1981.

[RS98] Alexander Russell and Ravi Sundaram. Symmetric alternation captures BPP. *Computational Complexity*, 7(2):152–162, 1998. doi:10.1007/s000370050007.

[RTVV99] Ran Raz, Gábor Tardos, Oleg Verbitsky, and Nikolai Vereshchagin. Arthur–Merlin games in boolean decision trees. *Journal of Computer and System Sciences*, 59(2):346–372, 1999. doi:10.1006/jcss.1999.1654.

[She13] Alexander Sherstov. The intersection of two halfspaces has high threshold degree. *SIAM Journal on Computing*, 42(6):2329–2374, 2013. doi:10.1137/100785260.

[She18] Alexander Sherstov. Breaking the Minsky–Papert barrier for constant-depth circuits. *SIAM Journal on Computing*, 47(5):1809–1857, 2018. doi:10.1137/15M1015704.

[Sip83] Michael Sipser. A complexity theoretic approach to randomness. In *Proceedings of the 15th Symposium on Theory of Computing (STOC)*, pages 330–335. ACM, 1983. doi:10.1145/800061.808762.

[van06] Dieter van Melkebeek. A survey of lower bounds for satisfiability and related problems. *Foundations and Trends in Theoretical Computer Science*, 2(3):197–303, 2006. doi:10.1561/0400000012.

[Ver92] Nikolai Vereshchagin. On the power of PP. In *Proceedings of the 7th Structure in Complexity Theory Conference (STRUCTURES)*, pages 138–143. IEEE, 1992. doi:10.1109/SCT.1992.215389.

[Ver95] Nikolai Vereshchagin. Lower bounds for perceptrons solving some separation problems and oracle separation of AM from PP. In *Proceedings of the 3rd Israel Symposium on Theory of Computing and Systems (ISTCS)*, pages 46–51. IEEE, 1995. doi:10.1109/ISTCS.1995.377047.

[Ver99] Nikolai Vereshchagin. Relativizability in complexity theory. In *Provability, Complexity, Grammars*, volume 192 of *AMS Translations, Series 2*, pages 87–172. American Mathematical Society, 1999.

[Vio09] Emanuele Viola. On approximate majority and probabilistic time. *Computational Complexity*, 18(3):337–375, 2009. doi:10.1007/s00037-009-0267-3.

[vW12] Dieter van Melkebeek and Thomas Watson. Time-space efficient simulations of quantum computations. *Theory of Computing*, 8(1):1–51, 2012. doi:10.4086/toc.2012.v008a001.

[Wil08] Ryan Williams. Time-space tradeoffs for counting NP solutions modulo integers. *Computational Complexity*, 17(2):179–219, 2008. doi:10.1007/s00037-008-0248-y.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il