Relations between Average-case and Worst-case Complexity

A. Pavan⋆[1] and N. V. Vinodchandran⋆⋆[2]

[1] Department of Computer Science
Iowa State University
email: pavan@cs.iastate.edu

[2] Department of Computer Science and Engineering
University of Nebraska-Lincoln
email: vinod@cse.unl.edu

Abstract. The consequences of the worst-case assumption NP = P are very well understood. On the other hand, we only know a few consequences of the average-case assumption "NP is easy on average." In this paper we establish several new results on the *worst-case complexity* of Arthur-Merlin games (the class AM) under the *average-case complexity* assumption "NP is easy on average."

We first consider a stronger notion of "NP is easy an average" namely NP is easy on average with respect to distributions that are computable by a polynomial size circuit family. Under this assumption we show:

- AM $\subseteq$ NSUBEXP.

Under the assumption that NP is easy on average with respect to polynomial-time computable distributions, we show:

- AME = E where AME is the exponential version of AM. This improves an earlier known result that if NP is easy on average then NE = E.
- For every $c > 0$, AM $\subseteq$ [io-pseudo$_{\mathrm{NTIME}(n^c)}$]$-$NP. Roughly this means that for any language L in AM there is a language L' in NP so that it is computationally infeasible to distinguish L from L'.

We use results from several sub-areas of complexity theory, including derandomization, for establishing our results.

1 Introduction

Can an average-case complexity collapse lead to the collapse of worst-case complexity classes? We explore this question in the framework introduced by Levin [Lev86].

In Levin's framework, a distributional problem is a decision problem A with a probability distribution on the instances of A. Such a distributional problem

⋆ Research supported in part by NSF grant CCF-0430807.
⋆⋆ Research supported in part by NSF grants CCF-0430991, CCR-0311577 and a Big 12 Faculty Fellowship.

is in Average-P if there is a deterministic algorithm that solves A in polynomial time on the average. DistNP is the set of pair (A, μ) where $A \in$ NP and μ is a polynomial-time computable distribution. The question analogous to NP $\stackrel{?}{=}$ P here is whether DistNP $\subseteq$ Average-P (in words NP is easy on average). Levin [Lev86] showed that there are distributional problems that are complete for DistNP in the sense that NP is easy on average if and only if the complete problem is solvable in Average-P. We refer the reader to [Gur91,Wan97] for pointers to many nice results in this area.

Understanding relations between average-case complexity and worst-case complexity is one of the most fundamental problems in complexity theory. In general we would like to answer the following question: "If a class $\mathcal{C}$ is easy on average, then what happens in the worst-case world?" This question has been studied extensively for the class E under various notions of average-case complexity. For example, it is now known that if E can be approximated by small circuits, then E indeed has small circuits [BFNW91,IW97]. Some problems such as the Permanent are shown to have a remarkable property—their worst-case complexity is essentially the same as their average-case complexity [Lip91,GS92,FL92]. Such proofs are based on the random-self reducibility of the Permanent. Similarly, random-self reducibility of PSPACE-complete problems can be used to show that if PSPACE is easy on average, then PSPACE = P [KS04].

Consequences of the assumption "NP is easy on average" have also been studied a lot. Ben-David, Chor, Goldreich, and Luby [BDCGL92] showed that if NP is easy on average, then E = NE. Impagliazzo [Imp95] showed that if NP is easy on average, then BPP = ZPP.

Köbler and Schuler [KS04] explored this connection further and showed that the assumption derandomizes MA to NP. Later Arvind and Köbler [AK02] showed in fact the assumption implies AM $\cap$ co-AM = NP $\cap$ co-NP. Buhrman, Fortnow, and Pavan [BFP03] showed that if NP is easy on average, then pseudorandom generators exist and BPP = P. They used this fact to show that a witness of any NE machine can be computed in E time. But it is still open whether the assumption that NP is easy on average implies derandomization of AM to NP. In this paper we explore this possibility.

We first consider a stronger notion of NP being easy on average. We show that if NP is easy on average with respect to distributions that are computable by polynomial-size circuits, then AM can be derandomized to nondeterministic subexponential time.

Under the assumption NP is easy with respect to polynomial-time computable distributions, we show that AME = E where AME is the exponential version of AM. This improves the earlier mentioned collapse result DistNP $\subseteq$ Average-P $\Rightarrow$ NE = E [BDCGL92]. In the polynomial-time range we show that DistNP $\subseteq$ Average-P implies AM is almost NP: for any language $L \in$ AM there is a language L' in NP so that L and L' are computationally indistinguishable infinitely often.

We use results from several subareas of complexity theory including derandomization to prove our results.

2 Preliminaries

We assume familiarity with definitions of standard complexity classes. Please refer to [BDG88,Pap94] for definitions of standard complexity classes.

An Arthur-Merlin game is a combinatorial game, played by Arthur–a probabilistic polynomial-time machine (with public coins), and Merlin–a computationally unbounded Turing machine.

Given a string x, Merlin tries to convince Arthur that x belongs to some language L. The game consists of a finite number of moves. The moves alternate between Arthur and Merlin. In each move Arthur (or Merlin) prints a finite string on a read-write communication tape. Arthur's moves can depend on his random bits. In the end Arthur either accepts or rejects x.

Babai and Moran [Bab85,BM88] define a language L to be in AM as follows. For every string x of length n, the game consists of a random move by Arthur and a reply by Merlin. If $x \in L$, then the probability that there exists a move by Merlin that leads to acceptance by Arthur is at least $\frac{3}{4}$; on the other hand, if $x \notin L$, then the probability that there exists a move by Merlin that leads to acceptance by Arthur is at most $\frac{1}{4}$.

$\mathrm{AMTIME}(l(n))$ denotes the class of languages accepted by a 2-round Arthur-Merlin interactive protocol with maximum message length $l(n)$ by both Arthur and Merlin. Let $\mathrm{NLIN} = \mathrm{NTIME}(O(n))$.

We need the exponential version of these complexity classes: $\mathrm{E} = \mathrm{DTIME}(2^{O(n)})$, $\mathrm{EXP} = \mathrm{DTIME}(2^{n^{O(1)}})$, $\mathrm{NEXP} = \mathrm{DTIME}(2^{n^{O(1)}})$, $\mathrm{AME} = \mathrm{AM}(2^{O(n)})$. For the nondeterministic case we will also need the sub-exponential version $\mathrm{NSUBEXP} = \cap_{\epsilon>0}\mathrm{NTIME}(2^{n^{\epsilon}})$.

Let $\Sigma = \{0,1\}$ and $A^n = \Sigma^n \cap A$. For any complexity class $\mathcal{C}$, the class ${}_{\mathrm{io}}\mathcal{C}$ is the class of languages $\{A \mid \exists B \in \mathcal{C}$ such that for infinitely many n, $A^n = B^n\}$.

Lemma 1. *For every $k > 0$, $\mathrm{E} \not\subseteq {}_{\mathrm{io}}\mathrm{DTIME}(2^{kn})$.*

2.1 Average-case complexity

We review some basic definitions from average-case complexity. We recommend Wang's survey [Wan97] for an excellent introduction to the area.

A *distributional problem* is a pair (A,μ) so that $A \subseteq \Sigma^*$ and μ is a probability distribution on Σ^*. DistNP is the class of distributional problems (A,μ) so that $A \in \mathrm{NP}$ and μ^* is polynomial time computable where μ^* denotes the distribution function of μ. Recall that $\mu^*(x) = \sum_{y \le x} \mu(y)$.

We need parameterized versions of the definitions of various notions in average-case complexity. In particular we will need the parameterized subclass of DistNP we call DistNLIN

Definition 1.

- $\mathrm{DistNLIN} = \{(A,\mu) \mid A \in \mathrm{NLIN}$ *and μ^* is linear-time computable*$\}$, *where μ^* denotes the distribution function of μ.*

- *Let μ be a distribution on Σ^*. Then a function $f : \Sigma^* \to \mathbf{N}$ is said to be n^k on μ-average [Lev86] if*

$$\sum_{x \in \Sigma^*} \frac{f^{1/k}(x)\mu(x)}{|x|} < \infty.$$

- *A distributional problem (A, μ) is in* AvgTIME(n^k) *if there exists a deterministic Turing machine M that accepts A so that the running time of M is n^k on μ-average. (A, μ) is in* Average-P *if there exists a k so that (A, μ) is in* AvgTIME(n^k).

By " NP is easy on average" we mean DistNP $\subseteq$ Average-P. Thus if DistNP $\subseteq$ Average-P, then for every language L in NP, and every polynomial-time computable distribution μ, $(L, \mu) \in$ Average-P.

Remark 1. Instead of working with distributions on Σ^*, we can also consider an ensemble of distributions $\mu = (\mu_1, \mu_2, \cdots)$, where μ_i is a distribution on Σ^i. An ensemble is polynomial-time computable, if there is a polynomial-time computable machine that on inputs i and x outputs $\mu_i^*(x)$. Impagliazzo [Imp95] defined a notion of average polynomial-time with respect to an ensemble of distributions. Though we state our results using distributions over Σ^*, all our results carry over to the case of ensemble of distributions.

Levin [Lev86] introduced the notion of reductions between distributional problems and defined DistNP-completeness. There are distributional problems that are DistNP-complete in the sense that NP is easy on average if and only if the complete problem is in Average-P. Proving distributional completeness is much more challenging than proving usual NP-completeness since the reductions must satisfy certain additional *domination* properties.

Definition 2. *([Lev86]) Let μ and ν be two distributions on Σ^*.*

- *We say that ν* dominates *μ* within *n^k if for all x, $\mu(x) \leq |x|^k \nu(x)$.*
- *Let $f : \Sigma^* \to \Sigma^*$. Then we say that ν* dominates *μ* within *n^k* via *f if there is a distribution μ_1 such that μ_1 dominates μ within n^k, and for all y in the range of f $\nu(y) = \sum_{y=f(x)} \mu_1(x)$.*
- *(A, μ)* reduces to *(B, ν) if there is a polynomial time computable many-one reduction f from A to B so that for some k, ν dominates μ within n^k via f*

Gurevich [Gur91] showed that the *distributional halting problem* (K, μ_K) is complete for DistNP, where $K = \{\langle i, x, 0^n \rangle \mid N_i \text{ accepts } x \text{ in } n \text{ steps } \}$, and $\mu_k(\langle i, x, 0^n \rangle) = \frac{1}{2^{|i|}|i|^2} \frac{1}{2^{|x|}|x|^2} \frac{1}{n^2}$. Here N_i denotes the i^{th} nondeterministic Turing machine in some fixed enumeration. We denote distributional halting problem by DH in this paper.

The following statement is easy to prove.

Observation 1 *If $(L, U) \in$ AvgTIME(n^k), then there is a deterministic Turing machine M that decides L and for all but finitely many n, there exist at most $2^n/n^2$ strings of length n, on which M takes more than n^{4k} time. Here U is the standard uniform distribution on Σ^*.*

Definition 3. *Two languages L and L' are* ae-NTIME(n^c)*-distinguishable, if there is a n^c-time bounded nondeterministic machine N such that for all but finitely many n, $M(0^n)$ outputs a string from $L \Delta L'$ along every accepting path. We say L and L' are* io-NTIME(n^c)*-indistinguishable, if they are not* ae-NTIME(n^c)*-distinguishable.*

Definition 4. *We say* AM $\subseteq$ [io-pseudo$_{\mathrm{NTIME}(n^c)}$]$-$NP*, if for every language L in* AM *there exists a language L' in* NP *such that L and L' are* io-NTIME(n^c)*-indistinguishable.*

3 Arthur-Merlin Games

We show several results on the worst-case complexity of Arthur-Merlin games under the assumption that NP is easy on average. First we consider a slightly stronger notion of NP is easy on average and prove that under this assumption AM $\subseteq$ NSUBEXP.

Let DistNP$_{\mathrm{P/poly}}$ denote the class of distributional problems (A, μ) so that $A \in$ NP and the distribution function μ^* is computable by a polynomial-size circuit family. We show that if DistNP$_{\mathrm{P/poly}} \subseteq$ Average-P, then AM $\subseteq$ NSUBEXP. We first show that the hypothesis implies E $\not\subseteq$ $_{\mathrm{io}}$(NP/`poly`). The result then follows from the following result of Shaltiel and Umans [SU01].

Theorem 1 ([SU01]). *If* E $\not\subseteq$ $_{\mathrm{io}}$(NP/`poly`)*, then* AM $\subseteq$ NSUBEXP*.*

The chain of arguments that we use is as follows. If E $\subseteq$ NP/`poly` and NP is easy on average then E $\subseteq$ Average-P/`poly` with respect to certain distribution (which is nonuniformly computable). Since there are random-self-reducible complete problems for E, we will actually get E $\subseteq$ P/`poly`. From [BFNW91] we have that E $\subseteq$ P/`poly` $\Rightarrow$ E = MA. Since NP is easy on average we have MA = NP [KS04] and NE = E [BDCGL92]. So we will finally get NE = NP which is a contradiction to the nondeterministic time hierarchy theorem. We now present a more formal argument.

We will need a non-uniform version of Average-P.

Definition 5. *A distributional problem (L, μ) is in* Average-P/`poly`*, if there is a Turing machine M and a polynomial-bounded function $a : \mathbb{N} \to \Sigma^*$, such that $x \in L \Leftrightarrow$ M accepts $\langle x, a(|x|)\rangle$, and there exists a constant $k > 0$, such that*

$$\sum_x \frac{T_M^{1/k}(\langle x, a(|x|)\rangle)}{|x|}\mu(x) < \infty.$$

where $T_M(y)$ denotes the running time of M on input y.

Note that the above definition requires the running time of M to be efficient on average only on correct advice.

We need the following lemma regarding random-self-reducibility of E complete problems.

Lemma 2 ([BFNW91]). *There exists a random-self-reducible complete language $L \in \mathrm{E}$ so that if there is a polynomial-size circuit C such that*

$$\forall n, \Pr_x[C(x) = L(x)] \geq 1 - 1/n^2,$$

then $L \in \mathrm{P}/\mathtt{poly}$.

We now prove the result we need to derandomize AM (actually we need an i.o. version of the following theorem. But we state and prove a cleaner non-i.o. version).

Theorem 2. *If* $\mathrm{DistNP}_{\mathrm{P/poly}} \subseteq \text{Average-P}$ *then* $\mathrm{E} \not\subseteq \mathrm{NP}/\mathtt{poly}$.

Proof. (*sketch*) Assume $\mathrm{E} \subseteq \mathrm{NP}/\mathtt{poly}$. Let L be a complete language for E that is random-self-reducible (provided by Lemma 2). Since $L \in \mathrm{NP}/\mathtt{poly}$, there is a language L' in NP and a polynomial-bounded advice function $a : \mathbb{N} \to \Sigma^*$, such that

$$\forall x, x \in L \Leftrightarrow \langle x, a(|x|)\rangle \in L'$$

Consider the following distribution μ

$$\mu(\langle x, y\rangle) = \begin{cases} U(x) & \text{if } y = a(|x|) \\ 0 & \text{otherwise} \end{cases}$$

Here U is the uniform distribution on Σ^*, i.e., $U(x) = \frac{1}{n^2}\frac{1}{2^n}$, where $n = |x|$. It is clear that μ is P/poly-computable. Since $\mathrm{DistNP}_{\mathrm{P/poly}} \subseteq \text{Average-P}$, $(L', \mu) \in \text{Average-P}$. Consider the following reduction from (L, U) to (L', μ): $f(x) = \langle x, a(|x|)\rangle$. It is clear that f is P/poly-computable and satisfies the dominance condition. Thus $(L, U) \in \text{Average-P}/\mathtt{poly}$.

Thus there is a Turing machine M and an advice function $a : \mathbb{N} \to \Sigma^*$, such that $x \in L \Leftrightarrow$ M accepts $\langle x, a(|x|)\rangle$, and there exists a constant l, such that

$$\sum_x \frac{T_M^{1/l}(\langle x, a(|x|)\rangle)}{|x|}\mu(x) < \infty.$$

By Observation 1, there is a constant k, such that for every n, on at least $(1 - 1/n^2)$ fraction of strings of the form $\langle x, a(|x|)\rangle$, M halts within n^k steps, and M accepts $\langle x, a(|x|)\rangle$ if and only if $x \in L$.

We now claim that this implies $L \in \mathrm{P/poly}$. Define a new machine M' as follows: M' on input $\langle x, y\rangle$ runs M on $\langle x, y\rangle$ for n^k steps. If M does not halt within n^k steps, then M' rejects its input. If M halts within n^k steps, then M' accepts $\langle x, y\rangle$ if and only if M accepts $\langle x, y\rangle$.

By converting M' into a circuit and hardwiring $a(|x|)$ in it, we obtain a polynomial-size circuit C such that

$$\forall n, \Pr[C(x) = L(x)] \geq 1 - 1/n^2.$$

Since L is random-self-reducible, by Lemma 2, $L \in \mathrm{P/poly}$. Since L is complete for E, $\mathrm{E} \subseteq \mathrm{P/poly}$. By [BFNW91], if $\mathrm{E} \subseteq \mathrm{P/poly}$, then $\mathrm{E} \subseteq \mathrm{MA}$. If $\mathrm{DistNP} \subseteq$

Average-P, then MA = NP [KS04] and also NE = E [BDCGL92]. Thus we have NE = NP. A contradiction follows from the nondeterministic time hierarchy theorem [Ž83].

Theorem 3. *If* $\mathrm{DistNP}_{\mathrm{P/poly}} \subseteq$ Average-P*, then* AM $\subseteq$ NSUBEXP*.*

Proof. (*Sketch*) Essentially the same proof will show that if $\mathrm{DistNP}_{\mathrm{P/poly}} \subseteq$ Average-P then $\mathrm{E} \not\subseteq {}_{\mathrm{io}}(\mathrm{NP}/\mathtt{poly})$. To see this, if $\mathrm{E} \not\subseteq {}_{\mathrm{io}}(\mathrm{NP}/\mathtt{poly})$ then we will get that E can be approximated by a polynomial size circuit on infinitely many input lengths. As argued in [BFNW91] we then get that $\mathrm{E} \subseteq {}_{\mathrm{io}}(\mathrm{P}/\mathtt{poly}) \subseteq {}_{\mathrm{io}}\mathrm{MA}$. If DistNP $\subseteq$ Average-P, then MA = NP [KS04], Thus we get $\mathrm{E} \subseteq {}_{\mathrm{io}}\mathrm{NP}$. Since a straightforward application of nondeterministic time hierarchy theorem is not sufficient to separate NE from ${}_{\mathrm{io}}\mathrm{NP}$ we need to argue slightly differently. By [BDCGL92], if DistNP $\subseteq$ Average-P, then E = NE. By a result of Impagliazzo, Kabanets, and Wigderson, if E = NE, then there is a fixed constant k such that $\mathrm{NTIME}(2^n) \subseteq \mathrm{DTIME}(2^{kn})$. Thus we obtain $\mathrm{E} \subseteq {}_{\mathrm{io}}\mathrm{NP} \subseteq {}_{\mathrm{io}}\mathrm{NTIME}(2^n) \subseteq {}_{\mathrm{io}}\mathrm{DTIME}(2^{kn})$. This contradicts Lemma 1.

Ben-David, Chor, Goldreich, and Luby [BDCGL92] showed that if DistNP $\subseteq$ Average-P, then E = NE. Buhrman, Fortnow, and Pavan showed that in fact a stronger conclusion follows, that is if DistNP $\subseteq$ Average-P, then for every NE-predicate $R(x, y)$, there is an E machine M such that $M(x)$ outputs a y such that $R(x, y)$ holds. Buhrman [Buh93] showed that if E = NE, and FSAT $\in \mathrm{PF}_{tt}^{\mathrm{NP}}$, then witnesses of NE predicates can be computed in E-time. Here FSAT denotes the problem of computing satisfying assignments of propositional formulas, and $\mathrm{PF}_{tt}^{\mathrm{NP}}$ is the class of functions that can be computed by polynomial-time machines that can make nonadaptive queries to an NP-oracle. This raises the following question: "If DistNP $\subseteq$ Average-P, is FSAT $\in \mathrm{PF}_{tt}^{\mathrm{NP}}$?". We have a partial answer.

Theorem 4. *If* $\mathrm{DistNP}_{\mathrm{P/poly}} \subseteq$ Average-P*, then* FSAT $\in \mathrm{SUBEXP}_{tt}^{\mathrm{NP}}$.

Proof. By Theorem 3, the hypothesis implies that $\mathrm{E} \not\subseteq {}_{\mathrm{io}}(\mathrm{NP}/\mathtt{poly})$. Very recently, Shaltiel and Umans [SU04] showed that $\mathrm{E} \subseteq \mathrm{NP}/\mathtt{poly}$ if and only if $\mathrm{E} \subseteq \mathrm{P}_{tt}^{\mathrm{NP}}/\mathtt{poly}$. Thus the hypothesis implies the existence of a language in E that does not have $\mathrm{P}_{tt}^{\mathrm{NP}}$-circuits. Klivans and van Melkebeek [KvM02] showed that there exist pseudo-random generators with polynomial stretch that are secure against $\mathrm{P}_{tt}^{\mathrm{NP}}$-circuits. These pseudo-random generators can be used to derandomize $\mathrm{BPP}_{tt}^{\mathrm{NP}}$ to $\mathrm{SUBEXP}_{tt}^{\mathrm{NP}}$. Since FSAT $\in \mathrm{BPP}_{tt}^{\mathrm{NP}}$ [VV85], we have the conclusion.

Now we show that, under the assumption DistNP $\subseteq$ Average-P, AME = E, and AM $\subseteq$ [io-pseudo$_{\mathrm{NP}}$]$-$NP. We also give an alternate proof of the result that under the assumption AM $\cap$ co-AM = NP $\cap$ co-NP due to Arvind and Köbler [AK02].

We use uniform derandomization results for AM and AM $\cap$ co-AM due to Gutfreud, Shaltiel, and Ta-Shma [GSTS03], and the fact that pseudorandom generators exist if NP is easy on average [BFP03] to prove our results.

Theorem 5 ([GSTS03]). *If* $\mathrm{E} \not\subseteq \mathrm{AMTIME}(2^{\beta n})$ *for some constant* β *then for every* $c > 0$*,* $\mathrm{AM} \subseteq [\text{io-pseudo}_{\mathrm{NTIME}(n^c)}]\text{--}\mathrm{NP}$.

Theorem 6 ([BFP03]). *If* $\mathrm{DistNP} \subseteq \text{Average-P}$ *then there is an algorithm that on input* 1^n *outputs a pseudorandom set for circuits of size* n*. The algorithm runs in time* n^c *for a fixed* c*.*

Recall that a set S is a pseudorandom set for circuits of size $s(n)$, if for every circuit C of size $s(n)$ if $\Pr_{x \in \Sigma^n}[C(x) = 1]$ is close to $\Pr_{x \in S}[C(x) = 1]$.

Our approach is as follows. Under the assumption $\mathrm{DistNP} \subseteq \text{Average-P}$, we show that $\mathrm{AMTIME}(2^n)$ is a subset of $\mathrm{DTIME}(2^{kn})$ for a fixed k. The same argument will also show that ${}_{\mathrm{io}}\mathrm{AMTIME}(2^n) \subseteq {}_{\mathrm{io}}\mathrm{DTIME}(2^{kn})$. Since for any fixed k, $\mathrm{E} \not\subseteq {}_{\mathrm{io}}\mathrm{DTIME}(2^{kn})$ we get that $\mathrm{E} \not\subseteq {}_{\mathrm{io}}\mathrm{AMTIME}(2^n)$ and therefore $\mathrm{AM} \subseteq \mathrm{Pseudo}_{\mathrm{NTIME}(n^c)}\text{--}\mathrm{NP}$ from Theorem 5.

For showing that $\mathrm{AMTIME}(2^n) \subseteq \mathrm{DTIME}(2^{kn})$ for a fixed k, we first observe a general result that if $\mathrm{DistNP} \subseteq \text{Average-P}$ then DistNLIN is in $\mathrm{AvgTIME}(n^k)$ for a fixed k. We use this result to first show that every tally language in $\mathrm{AMTIME}(n)$ is in $\mathrm{BPTIME}(n^k)$ for a fixed k. Finally we use the pseudorandom generator from Theorem 6 to get $\mathrm{BPTIME}(n^k) \subseteq \mathrm{DTIME}(n^l)$. A standard padding gives the collapse in the exponential level.

We know that if NP = P then for any k there is an l so that $\mathrm{NTIME}(n^k) \subseteq \mathrm{DTIME}(n^l)$. One way to see this is to observe that any problem in $\mathrm{NTIME}(T(n))$ is $T^2(n)$ time reducible to SAT and the result follows since under the assumption SAT is in $\mathrm{DTIME}(n^{k'})$ for a fixed k'. Such a theorem exists in the average-case setting also. We use the completeness of the Distributional Halting problem [Gur91] to prove an analogous result in the average-case setting. An accessible proof can be found in [Wan97].

Lemma 3 ([Lev86,Gur91]). *Let* (A, μ) *and* (B, ν) *be two distributional problems and let* f *be a reduction from* A *to* B *so that:*

- f *is computable in time* n^l
- ν *dominates* μ *within* n^r *via* f

Then if $B \in \mathrm{AvgTIME}(n^k)$ *then* $A \in \mathrm{AvgTIME}(n^{2klr})$*.*

Lemma 4 ([Lev86,Gur91]). *Every distributional problem* $(A, \mu) \in \mathrm{DistNLIN}$ *is reducible to the* DistNP *complete problem* $\mathrm{DH} = (K, \mu_K)$ *via a reduction* f_A *so that:*

- f_A *is computable in time* n^3
- μ_K *dominates* μ *within* n^3 *via* f_A

Using the above two lemmas we get the following theorem.

Theorem 7. *If* $\mathrm{DistNP} \subseteq \text{Average-P}$ *then there exists a* k *so that* $\mathrm{DistNLIN} \subseteq \mathrm{AvgTIME}(n^k)$*.*

Now we prove our main Lemma.

Lemma 5. *If* DistNP $\subseteq$ Average-P *then there exists a* k *so that* AMTIME$(2^n) \subseteq$ DTIME(2^{kn}).

Proof. (*sketch*) We show that under the assumption any tally set in AMTIME(n) is in DTIME(n^l) for a fixed l. The Lemma follows from the fact that for any language $L \in$ AMTIME(2^n), the tally version is in AMTIME(n).

Let L be a tally language in AMTIME(n). Then there is a language $B \in$ NLIN so that

$$0^n \in L \Rightarrow \Pr_{y \in \Sigma^n}[\langle 0^n, y\rangle \in B] = 1$$
$$0^n \notin L \Rightarrow \Pr_{y \in \Sigma^n}[\langle 0^n, y\rangle \in B] < 1/4$$

Consider the linear-time computable distribution μ which on a string $\langle 0^n, y\rangle$ has a probability of $\frac{1}{n^2 2^{|y|}}$. The distributional problem (B, μ) is in DistNLIN and hence is in AvgTIME(n^k) for a fixed k. Let M be the deterministic machine that witnesses this fact. Now consider a n^{4k} time deterministic machine M' that simulates M for n^{4k} steps. An easy counting argument shows that for every n, M' correctly decides B on at least $1 - 1/n^2$ fraction of strings of the form $\langle 0^n, y\rangle$. From this it easily follows that $L \in$ BPTIME(n^{4k}).

By Theorem 6, if DistNP $\subseteq$ Average-P, there is an n^c time-bounded algorithm that outputs a pseudorandom set for circuits of size n. Using this we can derandomize BPTIME(n^{4k}) to DTIME(n^{8k^2c}). Thus L is in DTIME(n^l) for $l = 8k^2c$.

Our theorems follow from the above Lemma.

Theorem 8. *If* DistNP $\subseteq$ Average-P, *then* AME = E.

Theorem 9. *If* DistNP $\subseteq$ Average-P *then,* AM $\subseteq$ [io-pseudo$_{\mathrm{NP}}$]$-$NP *for every* $c > 0$.

Proof. Proof follows from Lemma 5 and Theorem 5.

Our approach gives a different proof of the following result due to Arvind and Köbler [AK02].

Theorem 10 ([AK02]). *If* DistNP $\subseteq$ Average-P, *then* AM $\cap$ co-AM = NP $\cap$ co-NP.

Proof. The same argument as in Lemma 5 shows that if DistNP $\subseteq$ Average-P, then $_{\mathrm{io}}$AMTIME$(2^n) \subseteq {}_{\mathrm{io}}$DTIME$(2^{kn})$ for a fixed k. By Lemma 1, E $\not\subseteq {}_{\mathrm{io}}$DTIME$(2^{kn})$ for any fixed k. Therefore E $\not\subseteq {}_{\mathrm{io}}$AMTIME$(2^n)$. Gutfreud, Shaltiel, and Tashma [GSTS03] showed that this implies AM $\cap$ co-AM = NP $\cap$ co-NP.

Acknowledgments

We thank V. Arvind and Johannes Köbler for helpful discussions. We thank Chris Bourke and the FCT referees for useful comments.

References

[AK02] V. Arvind and J. Köbler. New lowness results for ZPP^{NP} and other complexity classes. *Journal of Computer and System Sciences*, 65(2):257–277, 2002.

[Bab85] L. Babai. Trading group theory for randomness. In *Proc. 17th Annual ACM Symp. on Theory of Computing*, pages 421–429, 1985.

[BDCGL92] S. Ben-David, B. Chor, O. Goldreich, and M. Luby. On the theory of average case complexity. *Journal of Computer and System Sciences*, 44(2):193–219, 1992.

[BDG88] J. Balcázar, J. Diaz, and J. Gabarró. *Structural Complexity I.* Springer-Verlag, Berlin, 1988.

[BFNW91] L. Babai, L. Fortnow, N. Nisan, and A. Wigderson. *BPP* has subexponential time simulations unless *exptime* has publishable proofs. In *Proceedings of the 6th Annual Conference on Structure in Complexity Theory, 1991*, pages 213–219, 1991.

[BFP03] H. Buhrman, L. Fortnow, and A. Pavan. Some results on derandomization. In *Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science*, volume LNCS 2607, pages 212–222, 2003.

[BM88] L. Babai and S. Moran. Arthur-merlin games: a randomized proof system, and a hierarchy of complexity class. *Journal of Computer and System Sciences*, 36(2):254–276, 1988.

[Buh93] H. Buhrman. *Resource bounded reductions*. PhD thesis, University of Amsterdam, 1993.

[FL92] U. Feige and C. Lund. On the hardness of computing permanent of random matrices. In *Proceedings of 24th Annual ACM Symposium on Theory of Computing*, pages 643–654, 1992.

[GS92] P. Gemmel and M. Sudan. Higly resilient correctors for polynomials. *Information Processing Letters*, 43:169–174, May 1992.

[GSTS03] D. Gutfreund, R. Shaltiel, and A. Ta-Shma. Uniform hardness vs. randomness tradeoffs for Arth ur-Merlin games. *Computational Complexity*, 12:85–130, 2003.

[Gur91] Y. Gurevich. Average case completeness. *Journal of Computer and System Sciences*, 42:346–398, 1991.

[Imp95] R. Impagliazzo. A personal view of average-case complexity theory. In *Proceedings of the 10th Annual Conference on Structure in Complexity Theory*, pages 134–147. IEEE Computer Society Press, 1995.

[IW97] R. Impagliazzo and A. Wigderson. P = BPP if E requires exponential circuits: Derandomizing the XOR lemma. In *Proceedings of the 29th ACM Symposium on Theory of Computing*, pages 220–229, 1997.

[KS04] J. Köbler and R. Schuler. Average-case intractability vs. worst-case intractability. *Information and Computation*, 190(1):1–17, 2004.

[KvM02] A. Klivans and D. van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM Journal on Computing*, 31:1501–1526, 2002.

[Lev86] L. Levin. Average case complete problems. *SIAM Journal of Computing*, 15:285–286, 1986.

[Lip91] R. Lipton. New directions in testing. In *Distributed Computing and Cryptography*, volume 2 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 191–202. American Mathematics Society, 1991.

[Pap94] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[SU01] R. Shaltiel and C. Umans. Simple extractors for all min-entropies and a new pseudo-random generator. In *42nd IEEE Symposium on Foundations of Computer Science*, pages 648–657, 2001.

[SU04] R. Shaltiel and C. Umans. Pseudorandomness for approximate counting and sampling. Technical Report TR 04-086, ECCC, 2004.

[VV85] L. Valiant and V. Vazirani. NP is as easy as detecting unique solutions. In *Proc. 17th ACM Symp. Theory of Computing*, pages 458–463, 1985.

[Ž83] S. Žák. A Turing machine time hierarchy. *Theor. Computer Science*, 26:327–333, 1983.

[Wan97] J. Wang. Average-case computational complexity theory. In L. Hemaspaandra and A. Selman, editors, *Complexity Theory Retrospective II*, pages 295–328. Springer-Verlag, 1997.