A Zero-one Law for RP and Derandomization of AM if NP is not Small

Philippe Moser *and Russell Impagliazzo †

Abstract

We show that if RP does not have p-measure zero then ZPP = EXP. As corollaries we obtain a zero-one law for RP in EXP, and that both probabilistic classes ZPP and RP have the same measure in EXP. We also prove that if NP does not have p-measure zero then NP = AM.

1 Introduction

Lutz resource-bounded measure [11], is a general framework that endows the complexity class EXP with a size notion (called $\mu(\cdot|\mathsf{EXP})$-measure), allowing to quantify the size of various subclasses of EXP such as $\mathsf{BPP}, \mathsf{NP}, \ldots$, where any class can be either small, large or in between, corresponding to having $\mu(\cdot|\mathsf{EXP})$-measure $0, 1$ or being non-measurable. Intuitively, a subclass of EXP has $\mu(\cdot|\mathsf{EXP})$-measure zero, if there is an exponential time computable predictor (a martingale) that can predict any language in the class without making too many mistakes.

Among many applications, it was used in derandomization, where van Melkebeek showed in [14] that BPP has either $\mu(\cdot|\mathsf{EXP})$-measure zero or one, thus ruling out the non-measurable case, and yielding further evidence that if randomness is not intractable (i.e. BPP $\neq$ EXP) then randomness is somehow easy (i.e. BPP has $\mu(\cdot|\mathsf{EXP})$-measure zero). It was also shown in [14] that the zero-one law holds for any subclass of BPP that is closed under polynomial-time truth-table reductions. Since RP is closed under truth-table reductions if and only if it is closed under complement, it was still left open whether RP also satisfies the zero-one law, since it is still not known whether RP is closed under complement. We show that this is the case, by proving a more general result: we show that if RP is not small in EXP then the smaller class ZPP is already intractable (i.e. ZPP = EXP). As Corollaries we obtain a zero-one law for RP in EXP, and that both probabilistic classes ZPP and RP have the same $\mu(\cdot|\mathsf{EXP})$-measure.

Resource bounded-measure can also be used in the theory of derandomization to formulate plausible hypothesis implying derandomization results. As an example, Lutz asked in [12] what derandomization results could be derived from the plausible hypothesis NP *does not have p-measure zero* (p-measure is the notion used to define measure in E), motivated by the intuition that NP not being small might imply it has enough computational power to derandomize its probabilistic version known as Arthur-Merlin games (AM). Arvind and Köbler proved in [2] that under this hypothesis, partial derandomization of AM was possible. More precisely they proved that NP not having p-measure zero implies $\mathsf{AM} \subset \mathsf{NP}/\log n$, where $\mathsf{NP}/\log n$ denotes non-uniform NP with logarithmic size advice. Using different techniques, we obtain full derandomization of AM under the same assumption, i.e. we show that NP does not have p-measure zero implies NP = AM.

As observed in [16] both our result also hold for the measure on the smaller (than EXP) complexity class SUBEXP, corresponding to subexponential time computable martingales instead of exponential ones, thus strengthening the statement "RP is small", and weakening the hypothesis "NP is small".

*Department of Computer Science, National University of Ireland Maynooth, Maynooth, Co. Kildare, Ireland.
†Computer Science and Engineering Department UC, San Diego

Both our proofs are inspired by the so-called "easy-witness" technique, that was first used in [7] to show an "easy or hard" result for RP, and that was consequently adapted in [10] to prove a similar result for NP. Further work along these lines can be found in [6, 3].

A draft of this paper was published in [4]; The proof given here is more direct, i.e. it does not not require van Melkebeek's zero-one law for BPP [14], but relies on results on the completeness of various sets of random strings from [1]. For similar results in the Baire categories setting, see [15].

2 Preliminaries

We use standard notation for traditional complexity classes; see for instance [17]. Let us fix some notations for strings and languages. Let $s_0, s_1, \ldots$ be the standard enumeration of the strings in $\{0,1\}^*$ in lexicographical order, where $s_0 = \lambda$ denotes the empty string. For any string x, $x-1$ denotes the predecessor of x. Denote by $s_0^{(n)}, s_1^{(n)}, \cdots, s_{2^n-1}^{(n)}$ all strings of size n ordered lexicographically. A sequence is an element of $\{0,1\}^\infty$. If w is a string or a sequence and $0 \leq i < |w|$ then $w[i]$ and $w[s_i]$ denotes the ith bit of w. Similarly $w[i \ldots j]$ and $w[s_i \ldots s_j]$ denote the ith through jth bits. We identify language L with its characteristic function χ_L, where χ_L is the sequence such that $\chi_L[i] = 1$ iff $s_i \in L$. $L \upharpoonright s_n$ stands for $L[s_0 \cdots s_n]$. Note that for any string x, $|L \upharpoonright x| = 2^{O(|x|)}$. If w_1 is a string and w_2 is a string or a sequence extending w_1, we write $w_1 \sqsubseteq w_2$. Denote by $L^{(k)}_{=n} = L \cap \{s_0^{(n)}, \cdots s_{k-1}^{(n)}\}$. A language L is said polynomially dense if there exists a polynomial p, such that $|L_{=n}| \geq 2^n/p(n)$, where $L_{=n}$ denotes the set of strings of size n contained in L. We sometimes write E_1 for E and E_2 for EXP.

2.1 Resource-bounded measure

In this section we describe the fragment of Lutz's measure theory for the class E and EXP that we will need. For a more detailed presentation of this theory we refer the reader to the survey by Lutz [13].

$\mu(\cdot|\mathsf{EXP})$-measure is obtained by imposing appropriate resource-bounds on a game theoretical characterization of the classical Lebesgue measure.

A martingale is a function $d : \{0,1\}^* \to [0, \infty[$ such that,

$$d(w) = \frac{d(w0) + d(w1)}{2}$$

for every $w \in \{0,1\}^*$. d is a p-martingale (sometimes written p_1) if d is computable in time polynomial in $|w|$ (or equivalently, $2^{O(|x|)}$ on input $w = L \upharpoonright x$). d is a p_2-martingale if d is computable in time $|w|^{\log^{O(1)}|w|}$ (or equivalently, $2^{|x|^{O(1)}}$ on input $w = L \upharpoonright x$).

This definition can be motivated by the following betting game in which a gambler puts bets on the successive membership bits of a hidden language A. The game proceeds in infinitely many rounds where at the end of round n, it is revealed to the gambler whether $s_n \in A$ or not. The game starts with capital 1. Then, in round n, depending on the first $n-1$ outcomes $w = \chi_A[0 \ldots n-1]$, the gambler bets a certain fraction $\epsilon_w d(w)$ of his current capital $d(w)$, that the nth word $s_n \in A$, and bets the remaining capital $(1-\epsilon_w)d(w)$ on the complementary event $s_n \notin A$. The game is fair, i.e. the amount put on the correct event is doubled, the one put on the wrong guess is lost. The value of $d(w)$, where $w = \chi_A[0 \ldots n]$ equals the capital of the gambler after round n on language A. The player wins on a language A if he manages to make his capital arbitrarily large during the game. We say that a martingale d succeeds on a language A, if $d(A) := \limsup_{w \sqsubset A, w \to A} d(w) = \infty$, where we identify language A with its characteristic sequence χ_A. The success set $S^\infty[d]$ of a martingale d is the class of all languages on which d succeeds.

For the rest of this section, let $i \in \{0,1\}$.

Definition 1 *A class C has p_i-measure zero if there is a single p_i-martingale d that succeeds on every language A of C.*

This property is monotone in the following sense: If class D is contained in a class C of p_i-measure zero, then D also has p_i-measure zero. It is easy to see that if a class C has p_1-measure zero, then it has p_2-measure zero (the converse is not always true).

Definition 2 *A class C has $\mu(\cdot|\mathsf{E}_i)$-measure zero (equivalently measure zero in E_i or $\mu(C|\mathsf{E}_i) = 0$) if $C \cap \mathsf{E}_i$ has p_i-measure zero. A class C has $\mu(\cdot|\mathsf{E}_i)$-measure one (equivalently measure one in E_i or $\mu(C|\mathsf{E}_i) = 1$) if its complement has $\mu(\cdot|\mathsf{E}_i)$-measure zero.*

Lutz showed in [11] that the classes E_i do not have $\mu(\cdot|\mathsf{E}_i)$-measure zero, which he called the measure conservation property.

Lutz also proved in [11] that uniform infinite unions of null classes are null.

Theorem 1 (Lutz) *Suppose $\{d_j\}_{j\geq 1}$ is a set of p-martingales, each succeeding on class C_j; where $d(j, w) := d_j(w)$ is computable in time $q(j, |w|)$ for a certain polynomial q. Then $\cup_{j\geq 1} C_j$ has p-measure zero.*

It is known from [18] that for any closed under symmetric difference (or closed under finite union and intersection) class C, $\mu(C|\mathsf{E}_i) = 1$ implies that $\mathsf{E}_i \subseteq C$.

2.2 Pseudorandom generator

We need the following definition of the relativized hardness of a pseudorandom generator.

Definition 3 *Let A be any language. The hardness $H^A(G_{m,n})$ of a random generator $G_{m,n} : \{0,1\}^m \longrightarrow \{0,1\}^n$, is defined as the minimal s such that there exists an n-input circuit C with oracle gates to A, of size at most s, for which:*

$$|\Pr_{x\in\{0,1\}^m}[C(G_m(x)) = 1] - \Pr_{y\in\{0,1\}^n}[C(y) = 1]| \geq \frac{1}{s}.$$

Klivans and van Melkebeek [8] noticed that Impagliazzo and Widgerson's [5] pseudorandom generator construction relativizes; i.e. for any language A, there is a deterministic polynomial time procedure that converts the truth table of a Boolean function that is hard to compute for circuits having oracle gates for A, into a pseudorandom generator that is pseudorandom for circuits with A oracle gates. More precisely,

Theorem 2 (Klivans-van Melkebeek [8]) *Let A be any language. There is a polynomial-time computable function $F : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$, with the following properties. For every $\epsilon > 0$, there exists $a, b \in \mathbb{N}$ such that*

$$F : \{0,1\}^{n^a} \times \{0,1\}^{b\log n} \to \{0,1\}^n,$$

and if r is the truth table of an $(a \log n)$-variables Boolean function of A-oracle circuit complexity at least $n^{\epsilon a}$, then the function $G_r(s) = F(r, s)$ is a generator, mapping $\{0,1\}^{b\log n}$ into $\{0,1\}^n$, which has hardness $H^{\mathrm{A}}(G_r) > n$.

2.3 Resource-Bounded Kolmogorov Complexity

Let us give the basic notions on resource-bounded Kolmogorov complexity that we will need.

Definition 4 *(Levin) [9] Let U be a universal Turing machine. Define* $\mathrm{Kt}(x)$ *to be* $\min\{|d|+\log t : U(d) = x$ *in at most t steps*$\}$.

It was shown in [1] (based on a result from [1] and an observation from Rahul Santhanam mentioned in [1]), that the existence of a polynomially dense set in P that contains only strings of high Kt-complexity implies $\mathsf{ZPP} = \mathsf{EXP}$.

Theorem 3 *[1] Let $0 < \delta < 1$, and suppose there is a set $R \in \mathsf{P}$ of polynomial density (for almost every length) such that $r \in R$ implies* $\mathrm{Kt}(r) \geq |r|^\delta$. *Then* $\mathsf{ZPP} = \mathsf{EXP}$.

3 A zero-one law for RP in EXP

Theorem 4 *Suppose* RP *does not have p-measure zero, then there exists a polynomially dense set* $R \in \mathsf{P}$*, and* $0 < \delta < 1$ *such that for every* $r \in R$*,* $\mathrm{Kt}(r) > |r|^{\delta}$*.*

Proof. To prove Theorem 4, we need the following lemma.

Lemma 1 *If* RP *does not have p-measure zero, then there exist a language* L *in* RP*, and a probabilistic polynomial time Turing machine* M *deciding* L*, such that for all but finitely many* $m \in \mathbb{N}$

1. $L^{(m)}_{=m}$ *is non-empty*
2. *For every* $x \in L^{(m)}_{=m}$*, every probabilistic witness* t *such that* $M(x,t) = 1$ *has large* Kt *complexity, i.e.* $\mathrm{Kt}(t) > |x|/2$*.*

Proof. Consider the following martingale d that allocates capital 2^{-n} to bet on strings of length n according to the following strategy (i.e. d total initial capital is $\sum_{i\geq 1} 2^{-i} = 1$). For every string's size n, d only bets on the n strings $s^{(n)}_0$ to $s^{(n)}_{n-1}$. On input $L \upharpoonright s^{(n)}_i$ with $i < n$, d checks that there is no element of L in the range $s^{(n)}_0$ to $s^{(n)}_{i-1}$, and if so d wagers $2^i/2^n$ that $s^{(n)}_i \notin L$, else d stops betting on n-sized strings. If $s^{(n)}_i$ is the first element of L in the range (i.e $L(s^{(n)}_i) = 1$), d loses

$$|\sum_{j=0}^{i-1} \frac{2^j}{2^n} - \frac{2^i}{2^n}| = 1/2^n$$

i.e. exactly the capital it allocated for strings of size n. On the other hand, each time there is no $x \in L$ between $s^{(n)}_0$ and $s^{(n)}_{n-1}$, d wins $2^n/2^n = 1$. Thus, if this happens infinitely often, d's wins grow arbitrarily large, whence the set $C_1 = \{L|\ \exists^{\infty} n : L^{(n)}_{=n} = \emptyset\}$ satisfies $C_1 \subseteq S^{\infty}[d]$. It is easy to see that d can be computed in $2^{O(n)}$ steps.

Consider the following martingale d' that only bets on the n first n-sized strings $s^{(n)}_0$ to $s^{(n)}_{n-1}$, for every length n. Fix an enumeration of polytime probabilistic Turing machines so that M_j on any input z always halts within $|z|^{\log j}$ steps, and a deterministic universal Turing machine U. On input $L \upharpoonright s^{(n)}_i$ with $0 \leq i \leq n-1$, d' simulates the first $\log n$ probabilistic machines on all probabilistic witnesses for inputs of length at most $\log n$. Let M_j be the first such machine that agrees with L on all such inputs, making errors only when $x \in L$, and then on less than $1/4$ of its probabilistic witness, i.e. decides L correctly (RP-wise) on all inputs smaller than $\log n$. d' simulates U on all programs p of size smaller than $n/2$, during $2^{n/2}$ steps. For every string t output during U's simulation, it checks whether $M_j(s^{(n)}_i, t) = 1$. If there is such a t (denote this by $S(s^{(n)}_i) = 1$, i.e. the simulation yielded 1), d' bets half its current capital that $s^{(n)}_i \in L$. Otherwise, d' does not bet. Thus, for all n and $0 \leq k \leq 2^n - 1$

$$d'(L \upharpoonright s^{(n)}_k) = \begin{cases} 3d'(L \upharpoonright s^{(n)}_k - 1)/2 & \text{if } 0 \leq k \leq n-1,\ S(s^{(n)}_k) = L(s^{(n)}_k) = 1 \\ d'(L \upharpoonright s^{(n)}_k - 1)/2 & \text{if } 0 \leq k \leq n-1,\ S(s^{(n)}_k) = 1 \text{ and } L(s^{(n)}_k) = 0 \\ d'(L \upharpoonright s^{(n)}_k - 1) & \text{otherwise.} \end{cases}$$

Claim 1 *Let* C_2 *be the set of languages* $L \in$ RP *such that for the first probabilistic poly-time machine deciding* L *(denoted* M_{j_0}*), there are infinitely many lengths* m*, such that there exists* $x \in L^{(m)}_{=m}$ *and a probabilistic witness* t *with* $\mathrm{Kt}(t) < |x|/2$*, such that* $M_{j_0}(x,t) = 1$*. Then* $C_2 \subseteq S^{\infty}[d']$*.*

Let $L \in C_2$. After a finite amount of time d' will always pick $j = j_0$. Thereafter for each length m such that there exists $x \in L^{(m)}_{=m}$ and a probabilistic witness t with low Kt complexity, such that $M_{j_0}(x,t) = 1$, i.e. $S(x) = 1$ and $L(x) = 1$, $d'(L \upharpoonright x) = 3d'(L \upharpoonright x - 1)/2$, i.e. the capital is multiplied by $3/2$ (and d' never errs from now on). Therefore if there are infinitely many such lengths m, d' grows unbounded, hence the claim is proved.

The running time of d' is is less than $2^{O(n)}$, since on input $L \restriction s_i^{(n)}$ there are $\log n$ machines to simulate on all inputs of size $\log n$, where each machine runs in time smaller than $(\log n)^{\log\log n}$, i.e. takes time $2^{(\log n)^{\log\log n}}$ to be simulated by trying all probabilistic witnesses and taking a majority vote. Next d' needs to simulate U on $2^{n/2}$ programs during $2^{n/2}$ steps. Thus d' can be computed in time $2^{O(n)}$.

Let $C = C_1 \cup C_2$. By Theorem 1, C has p-measure zero. Therefore if RP does not have p-measure zero, $\mathsf{RP} \not\subseteq C$, which proves Lemma 1.

The proof of Theorem 4 then follows. Let L and M be as in Lemma 1 (denote by N the bound such that Lemma 1 holds for any length $n > N$), and suppose M runs in time n^k. Consider the set

$$R = \cup_{n>N}\{tv|\ t \in \{0,1\}^{n^k}, v \in \{0,1\}^*, |tv| < (n+1)^k,\ M(x,t) = 1 \text{ for some } x \in L^{(n)}_{=n}\}.$$

Because $L \in \mathsf{RP}$, we have $R \in \mathsf{P}$ and $R_{=n^k}$ is polynomially dense for every integer $n > N$. Let $m = n^k + l$ be any integer with $n^k < m < (n+1)^k$; since $|R_{=m}| = |R_{=n^k}| \cdot 2^l$, R is polynomially dense for every length $m > N$. Moreover if $z \in R$, (with $z = tv$, $|t| = n^k$ and $|tv| < (n+1)^k$) then $M(x,t) = 1$ for some $x \in L^{(n)}_{=n}$, therefore $\mathrm{Kt}(t) > |x|/2 = n/2$. Since $\mathrm{Kt}(tv) > \mathrm{Kt}(t) - O(\log n)$ (because any program for tv yields a program for t by producing tv and dropping v, with a extra complexity of at most $O(\log n)$), we have (for n large enough)

$$\mathrm{Kt}(tv) > \mathrm{Kt}(t) - O(\log n) > n/2 - O(\log n) > n^{1/2} = n^{\frac{k+1}{2(k+1)}} > |tv|^{\frac{1}{2(k+1)}}.$$

Putting $\delta := \frac{1}{2k+1}$ ends the proof.

The following result states that the zero-one law holds for RP in EXP.

Theorem 5 RP *has either* $\mu(\cdot|\mathsf{EXP})$*-measure zero or one.*

Proof. Suppose RP does not have $\mu(\cdot|\mathsf{EXP})$-measure zero i.e., it does not have p-measure zero; then by Theorem 4, P contains a polynomially dense set R of strings with high Kt complexity. From Theorem 3 we have $\mathsf{ZPP} = \mathsf{EXP}$ i.e., $\mathsf{RP} = \mathsf{EXP}$, thus $\mu(\mathsf{RP}|\mathsf{EXP}) = 1$

Corollary 1 ZPP *and* RP *have the same* $\mu(\cdot|\mathsf{EXP})$*-measure.*

4 Derandomization of AM if NP is not small

Let us show our second main result, stating that if NP does not have p-measure zero, then AM can be fully derandomized.

Theorem 6 *Suppose* NP *does not have p-measure zero. Then* $\mathsf{NP} = \mathsf{AM}$.

Proof. In order to prove Theorem 6, we need the following lemma.

Lemma 2 *If* NP *does not have p-measure zero, then there exist a language* L *in* NP *and a nondeterministic polynomial time Turing machine* M *deciding* L*, such that for all but finitely many* $m \in \mathbb{N}$

1. $L^{(m)}_{=m}$ *is non-empty*
2. *for every* $x \in L^{(m)}_{=m}$*, every nondeterministic witness* t *such that* $M(x,t) = 1$*, has large* SAT*-oracle circuit complexity, i.e.* $\mathrm{Size}^{\mathrm{SAT}}(t) > |x|^{1/2}$.

Proof. The proof is similar to Theorem 4. Suppose NP does not have p-measure zero; the first property is easily verified by constructing the same martingale d as in the proof of Theorem 1. For the second property consider the following martingale d' that only bets on the n first n-sized strings $s_0^{(n)}$ to $s_{n-1}^{(n)}$, for every length n. Fix an enumeration of nondeterministic Turing machines so that M_j on any input z always halts within $|z|^{\log j}$ steps. On input $L \restriction s_i^{(n)}$ with $0 \leq i \leq n$,

d' simulates the first $\log n$ nondeterministic machines on all nondeterministic witnesses for inputs of length at most $\log n$. Let M_j be the first such machine that agrees with L NP-wise on all such inputs, i.e. decides L correctly (NP-wise) on inputs smaller than $\log n$. Next d' constructs all circuits with oracle gates for SAT of size smaller than $n^{1/2}$ on $\log j \log n$ inputs, and computes the truth table t for each. If $M_j(s_i^{(n)}, t) = 1$ for any such t (denote this by $S(s_i^{(n)}) = 1$, i.e. the simulation yielded 1), d' bets half its current capital that $s_i^{(n)} \in L$. Otherwise, d' does not bet. Thus, for all n and $0 \leq k \leq 2^n - 1$

$$d'(L \upharpoonright s_k^{(n)}) = \begin{cases} 3d'(L \upharpoonright s_k^{(n)} - 1)/2 & \text{if } 0 \leq k \leq n-1,\ S(s_k^{(n)}) = L(s_k^{(n)}) = 1 \\ d'(L \upharpoonright s_k^{(n)} - 1)/2 & \text{if } 0 \leq k \leq n-1,\ S(s_k^{(n)}) = 1 \text{ and } L(s_k^{(n)}) = 0 \\ d'(L \upharpoonright s_k^{(n)} - 1) & \text{otherwise.} \end{cases}$$

Claim 2 *Let C_2 be the set of languages $L \in$ NP such that for the first nondeterministic poly-time machine deciding L (denoted M_{j_0}), there are infinitely many lengths m, such that there exists $x \in L^{(m)}_{=m}$ and a nondeterministic witness t that when viewed as a function of $\log |t|$ inputs has* SAT*-oracle circuit complexity less than $|x|^{1/2}$. Then $C_2 \subseteq S^\infty[d']$.*

After a finite amount of time d' will always pick $j = j_0$. Thereafter for each length m such that there exists $x \in L^{(m)}_{=m}$ and a nondeterministic witness t with small circuit complexity, such that $M_{j_0}(x, t) = 1$, i.e. $S(x) = 1$ and $L(x) = 1$, $d'(L \upharpoonright x) = 3d'(L \upharpoonright x - 1)/2$, i.e. the capital is multiplied by 3/2 (and d' never errs from now on). Therefore if there are infinitely many such lengths m, d' grows unbounded, hence the claim is proved.

The running time of d' is less than $2^{O(n)}$, since on input $L \upharpoonright s_i^{(n)}$ there are $\log n$ machines to simulate on all inputs of size $\log n$, where each machine runs in time smaller than $(\log n)^{\log \log n}$, i.e. takes time $2^{(\log n)^{\log \log n}}$ to be simulated by trying all nondeterministic witnesses. Next d' needs to construct 2^n circuits of size at most $n^{1/2}$ and construct their truth table. Since evaluating a SAT gate of size at most $n^{1/2}$ takes time $2^{O(n)}$, $d'(L \upharpoonright s_i^{(n)})$ can be computed in time $2^{O(n)}$.

Let $C = C_1 \cup C_2$. By Theorem 1, C has p-measure zero. Therefore if NP does not have p-measure zero, NP $\not\subseteq C$, which proves Lemma 2.

The proof of Theorem 6 then follows, let L and M be as in Lemma 2, where M runs in time n^d. Let $L' \in$ AM be any language and N a probabilistic nondeterministic Turing machine deciding it, and assume that on input of size n, N runs in time n^c. For $\epsilon = 1$ let a and b be as in Theorem 2 and pick m so that $m^{1/2} > n^e$, where $e = ac$. For every x among $s_0^{(m)}, \cdots, s_{m-1}^{(m)}$ nondeterministically guess a witness t for machine M on input x. Check whether $M(x, t) = 1$ − we know that there is at least one such witness for at least one such x, and every such witness has circuit complexity at least $m^{1/2} = n^{ac}$ − if so use Theorem 2 to construct from t a pseudorandom generator from $O(\log n)$ bits to n^c bits secure against circuits of size n^c with oracle gates to SAT. Use the outputs of this pseudorandom generator to simulate the nondeterministic Turing machine N for L'. This has expected nondeterministic polynomial time, and never errs for sufficiently large inputs, so $L' \in$ NP.

References

[1] Eric Allender, Harry Buhrman, Michal Koucký, Dieter van Melkebeek, and Detlef Ronneburger. Power from random strings. *SIAM J. Comput.*, 35(6):1467–1493, 2006.

[2] V. Arvind and J. Köbler. On pseudorandomness and resource-bounded measure. *Proceedings 17th Conference of the Foundations of Software Technology and Theoretical Computer Science*, 1346:235–249, 1997.

[3] John M. Hitchcock and Aduri Pavan. Hardness hypotheses, derandomization, and circuit complexity. In Kamal Lodaya and Meena Mahajan, editors, *FSTTCS*, volume 3328 of *Lecture Notes in Computer Science*, pages 336–347. Springer, 2004.

[4] R. Impagliazzo and P. Moser. A zero-one law for RP. *Proceedings of the 18th Conference on Computational Complexity*, pages 48–52, 2003.

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

[6] Russell Impagliazzo, Valentine Kabanets, and Avi Wigderson. In search of an easy witness: exponential time vs. probabilistic polynomial time. *J. Comput. Syst. Sci.*, 65(4):672–694, 2002.

[7] Valentine Kabanets. Easiness assumptions and hardness tests: Trading time for zero error. *J. Comput. Syst. Sci.*, 63(2):236–252, 2001.

[8] A. Klivans and D. van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial hierarchy collapses. *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 659–667, 1999.

[9] Leonid A. Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61(1):15–37, 1984.

[10] Chi-Jen Lu. Derandomizing arthur-merlin games under uniform assumptions. *Computational Complexity*, 10(3):247–259, 2001.

[11] J.H. Lutz. Almost everywhere high nonuniform complexity. *Journal of Computer and System Science*, 44:220–258, 1992.

[12] J.H. Lutz. Observations on measure and lowness for Δ_2^{p}. *Theory of Computing Systems*, 30:429–442, 1997.

[13] J.H. Lutz. The quantitative structure of exponential time. In L.A. Hemaspaandra and A.L. Selman, editors, *Complexity Theory Retrospective II*, pages 225–260. Springer, 1997.

[14] D. Melkebeek. The zero-one law holds for BPP. *Theoretical Computer Science*, 244(1-2):283–288, 2000.

[15] Philippe Moser. Baire categories on small complexity classes and meager-comeager laws. *Inf. Comput.*, 206(1):15–33, 2008.

[16] Philippe Moser. Martingale families and dimension in P. *Theor. Comput. Sci.*, 400(1-3):46–61, 2008.

[17] C. Papadimitriou. *Computational complexity.* Addisson-Wesley, 1994.

[18] K. Regan, D. Sivakumar, and J. Cai. Pseudorandom generators, measure theory and natural proofs. *In Proc. 36th Annual IEEE Symposium on Foundations of Computer Science*, pages 26–35, 1995.