On the Resource Bounded Measure of P/poly (Extended Abstract)

Johannes Köbler and Wolfgang Lindner
Theoretische Informatik, Universität Ulm, D-89069 Ulm, Germany

Abstract

We show that the class of sets having polynomial size circuits, P/poly*, has* $\mathrm{EXP}^{\mathrm{NP}}$*-measure zero under each of the following two assumptions:*

- $\mathrm{EXP}^{\mathrm{NP}} \neq \mathrm{ZPP}^{\Sigma_2^p}$ *(which holds if the polynomial time hierarchy does not collapse to* $\mathrm{ZPP}^{\Sigma_2^p}$*), or*
- NP *is not small (does not have* EXP*-measure zero).*

1 Introduction

A central issue in the study of resource bounded measure as introduced by Lutz [Lut92] is the measure of nonuniform complexity classes. Already in [Lut92] Lutz showed that the class of sets having polynomial-size circuits, P/poly, has EXPSPACE-measure zero. This means that *almost all* languages in EXPSPACE do not have polynomial-size circuits and hence, improves on the result due to Kannan [Kan82] that there *exists* a language in EXPSPACE which does not have polynomial-size circuits. Using Stockmeyer's approximation of $\#\mathrm{P}$ functions in polynomial time relative to an oracle in Σ_2^p [Sto85], Mayordomo [May94] could replace EXPSPACE in Lutz' above mentioned result by $\mathrm{EXP}^{\Sigma_2^p}$, that is, P/poly has $\mathrm{EXP}^{\Sigma_2^p}$-measure zero. On the other hand, Regan et al. [RSC95] showed that P/poly does *not* have EXP-measure zero, provided that (cryptographic) one-way functions with exponential security exist. This gives evidence that P/poly does not have EXP-measure zero.

Here we consider the measure of P/poly at $\mathrm{EXP}^{\mathrm{NP}}$, the intermediate level of the exponential time hierarchy between EXP and $\mathrm{EXP}^{\Sigma_2^p}$. We show that P/poly has $\mathrm{EXP}^{\mathrm{NP}}$-measure zero at $\mathrm{EXP}^{\mathrm{NP}}$ under the following condition: there exists a pseudorandom generator G which is computable in exponential time relative to an oracle in NP and which has polynomial security infinitely often against polynomial-size oracle circuits equipped with an oracle in NP. Similar ideas have already been used in [BvMR$^+$98]. By results shown in [Lip91, GL89, Yao82, NW94, BFNW93], such a generator can be obtained from the assumption $\mathrm{EXP}^{\mathrm{NP}} \not\subseteq \mathrm{P}^{\mathrm{NP}}/\mathrm{poly}$. Because $\mathrm{EXP}^{\mathrm{NP}} \subseteq \mathrm{P}^{\mathrm{NP}}/\mathrm{poly}$ implies that $\mathrm{EXP}^{\mathrm{NP}} \subseteq \mathrm{ZPP}^{\Sigma_2^p}$ [BH92, KW95], it follows that P/poly has $\mathrm{EXP}^{\mathrm{NP}}$-measure zero unless $\mathrm{EXP}^{\mathrm{NP}} = \mathrm{ZPP}^{\Sigma_2^p}$.

Furthermore, we consider the hypothesis that NP is not small, i.e., NP does not have EXP-measure zero. This hypothesis implies the existence of a generator computable in exponential-time relative to some oracle in NP which is even much stronger than what is needed in our proof [Lut96, AK97]. Thus, P/poly has $\mathrm{EXP}^{\mathrm{NP}}$-measure zero unless NP has EXP-measure zero.

2 Preliminaries

We use the binary alphabet $\Sigma = \{0, 1\}$. The cardinality of a finite set X is denoted by $||X||$ and the length of $x \in \Sigma^*$ by $|x|$. For two strings v and w, we use $v \sqsubseteq w$ to denote that v is a prefix of w. The characteristic function of a language $A \subseteq \Sigma^*$ is defined as $A(x) = 1$ if $x \in A$, and $A(x) = 0$ otherwise.

The definitions of complexity classes we consider like P, NP, E, EXP etc. can be found in standard books [BDG95, BDG90, Pap94]. By $\log$ we denote the function $\log x = \max\{1, \lceil \log_2 x \rceil\}$.

For a class C of sets and a class F of functions from 1^* to Σ^*, let C/F [KL80] be the class of sets A such that there is a set $B \in \mathrm{C}$ and a function $h \in \mathrm{F}$ such that for all $x \in \Sigma^*$,

$$x \in A \Leftrightarrow \langle x, h(1^{|x|}) \rangle \in B.$$

The function h is called an *advice function* for A.

A *supermartingale* is a function $d : \{0,1\}^* \to \mathrm{R}^+$ satisfying the *average law*

$$d(w0) + d(w1) \leq 2d(w)$$

for all $w \in \{0,1\}^*$. A supermartingale *succeeds* on a language $A \subseteq \{0,1\}^*$ if

$$\limsup_{l \to \infty} d(A[s_1 \cdots s_l]) = \infty$$

where $s_1 = \lambda, s_2 = 0, s_3 = 1, s_4 = 00, s_5 = 01, \ldots$ is the standard enumeration of $\{0,1\}^*$ in lexicographic order and $A[s_1 \cdots s_l] = A(s_1) \cdots A(s_l)$.

Resource bounded measure zero classes are defined by imposing a resource bound on the computation of a martingale succeeding on every language in the class. Viewing a martingale d as a function from finite binary strings w it is obvious to think of the computation of a martingale done by a transducer with input w and output $d(w)$, and so the resources required to compute $d(w)$ should be expressed in terms of the length of w. However, here we always regard a martingale as a function of initial segments $A[s_1 \cdots s_l]$ of languages A (rather than a function of arbitrary binary strings w). Complexity bounds on languages are usually expressed in terms of the length of strings. Consequently, we find it more convenient to state the resource bound in terms of the length of the largest string s_l in the domain of $A[s_1 \cdots s_l]$ rather than in terms of the length $l = |A[s_1 \cdots s_l]|$ (cf. [BvMR+98]).

So we define a supermartingale d to be EXP-*computable* if there is an oracle transducer M, and a polynomial p, such that for every language A, and every string s_l, M on input s_l and oracle $A^{\le s_l} = A \cap \{s_1, \ldots, s_l\}$ computes $d(A[s_1 \cdots s_l])$ in time $O(2^{p(n)})$ where $n = |s_l|$. A class of languages $\mathcal{C}$ has EXP-*measure zero* if there is a EXP-computable supermartingale d that covers $\mathcal{C}$, i.e., d succeeds on every language $A \in \mathcal{C}$. (The EXP-computable martingales defined here correspond to the p_2-computable martingales defined by Lutz.)

Lutz showed that EXP-measure zero classes can be regarded as negligible small compared to EXP. In particular, Lutz showed that infinite unions of EXP-measure zero classes have EXP-measure zero if the classes are coverd by an uniformly exponential-time computable enumeration of martingales. Ambos-Spies et al. [ATZ97] extended this to arbitrary (countable) unions, provided that the involved martingales are all computable in some single exponential time bound. We state this result for later reference.

Theorem 1. [ATZ97] *If $\mathcal{C}_1, \mathcal{C}_2, \ldots$ is a collection of classes such that for some fixed constant k, each $\mathcal{C}_i$ is covered by some martingale computable in time $O(2^{n^k})$, then the union $\bigcup \mathcal{C}_i$ has* EXP*-measure zero.*

The notion of EXP-measure zero can be naturally extended to relativized classes EXP^B in the following way.

For any oracle B, we say that a supermartingale d is EXP^B-computable if there exists an exponentially time-bounded transducer M such that for every language A, and every string s_l, M on input s_l and oracle $A^{\le s_l} \oplus B$ computes $d(A[s_1 \cdots s_l])$. A class $\mathcal{C}$ has EXP^B-measure zero if there is a EXP^B-computable supermartingale d that succeeds on every language in $\mathcal{C}$.

3 The Measure of P/poly

In the proof of our theorem we use the relativized version of some intermediate steps on the way to obtain from the assumption $\mathrm{EXP} \neq \mathrm{MA}$ subexponential-time simulations for BPP [BFNW93].

In particular, we will use the steps that lead from a language $L \notin \mathrm{P/poly}$ to the construction of a pseudorandom generator with polynomial security infinitely often against polynomial size circuits [Lip91, GL89, Yao82, NW94].

It is not hard to check that this construction relativizes with respect to the adversaries. That is, if L is not in P^A/poly then the resulting pseudorandom generator has polynomial security infinitely often against polynomial-size oracle circuits using oracle A. (For proofs we refer the reader also to [BFNW93, GNW95, Lub97].)

Theorem 2. *Assume that* $\mathrm{EXP}^{\mathrm{NP}} \not\subseteq \mathrm{P}^{\mathrm{NP}}$*/poly. Then* P*/poly has* $\mathrm{EXP}^{\mathrm{NP}}$*-measure zero.*

Proof (Sketch). By (the relativised version of) Theorem 1, it suffices to show that for every integer k, there is a supermartingale d_k which is computable in time $O(2^{\log^c n})$, has access to some oracle in NP, and which succeeds on all languages with circuits of size n^k. Notice that the constant c must not depend on k.

In the following, we will consider each length n separately. For each length n, we will focus on prefixes of characteristic sequences for the first m strings of length n, where

$$m(n) = n^k + 3\lceil \log n \rceil.$$

For notational convenience, let $x_{n,1}, \ldots, x_{n,m}$ denote the lexicographically first m strings of length n. Throughout the proof, we will use the symbol "w" to denote a function $w : \{x_{n,1}, \ldots, x_{n,m}\} \to \{0,1\}$ (equivalently, we may think of w as a binary sequence of length m), and the the symbol "v" to denote a prefix of some w (i.e., v denotes a binary sequence of length at most m). Even though w is not completely defined for all strings of length n, we say that w *has a circuit of size* s if w can be extended to some function $f : \{0,1\}^n \to \{0,1\}$ which is computable by a circuit of size s.

Now, for every v with $|v| \le m$, let $P_{k,n}(v)$ denote the conditional probability that some uniformly at random chosen $w \in \{0,1\}^m$ has a circuit of size n^k, given v is a prefix of w:

$$P_{k,n}(v) = \Pr_w[\ w \text{ has a circuit of size } n^k \ | v \sqsubseteq w].$$

By the choice of m, $P_{k,n}(\lambda) \le 2^{n^k - m} \le 1/n^3$, and for a function w having a circuit of size n^k, $P_{k,n}(w) = 1$.

Note that for all v with $|v| < m$, $P_{k,n}(v0) + P_{k,n}(v1) = 2P_{k,n}(v)$, that is, $P_{k,n}$ satisfies the average law.

By our assumption that $\mathrm{EXP}^{\mathrm{NP}} \not\subseteq \mathrm{P}^{\mathrm{NP}}/\mathrm{poly}$, there is a language L in $\mathrm{EXP}^{\mathrm{NP}}$ such that for every polynomial p there exist infinitely many n such that for every oracle circuit C of size $p(n)$, $L^{=n} \neq C^{SAT}$. By results shown in [BFNW93, GL89, Yao82, NW94], for any integer t, L can be transformed into a pseudorandom generator G_t which takes as input a seed s of length n, and outputs a string of length n^t with the following security property: for all polynomials p, there exist infinitely many n, such that for every n^t-input oracle circuit C of size at most $p(n)$,

$$|\Pr_s[C^{SAT}(G_t(s)) = 1] - \Pr_r[C^{SAT}(r)) = 1]| \leq n^{-t},$$

where s and r are chosen uniformly at random from $\{0,1\}^n$ and $\{0,1\}^{n^t}$, respectively. Furthermore, the generators G_t are uniformly computable by an oracle Turing machine equipped with some oracle in NP, which on input t and s runs in time exponential in n.

In the following we will use the generator G_t with security parameter $t = k+4$ to obtain an approximation of $P_{k,n}(v)$ with error bound $\gamma = 1/n^t \leq 1/(2m+1)n^3$.

For all v with $|v| \leq m$, let $\hat{P}_{k,n}(v)$ be the fraction

$$\frac{\|\{s \in \{0,1\}^n : \ vu_s \text{ has a circuit of size } n^k\}\|}{2^n}$$

where u_s is the string consisting of the first $m - |v|$ bits of the output of $G_t(s)$.

Since there is an oracle circuit of size $O(m)$ with access to some oracle in NP which can test whether a given w of length m has a circuit of size n^k, it follows that there exist infinitely many n such that for all v with $|v| \leq m$,

$$|P_{k,n}(v) - \hat{P}_{k,n}(v)| \leq \gamma.$$

Now, using the approximation $\hat{P}_{k,n}$ we define $d_{k,n}$ for all v with $1 \leq |v| \leq m$ by the following clauses:

(i) If $v = \lambda$, and $\hat{P}_{k,n}(\lambda) \leq 1/n^3 + \gamma$, let

$$d_{k,n}(\lambda) = \hat{P}_{k,n}(\lambda) + 2\gamma m.$$

(ii) If $v \neq \lambda$, and for all proper prefixes v' of v: $\hat{P}_{k,n}(v'0) + \hat{P}_{k,n}(v'1) \leq 2 \cdot \hat{P}_{k,n}(v') + 4\gamma$, then let

$$d_{k,n}(v) = \hat{P}_{k,n}(v) + 2\gamma(m - |v|).$$

(iii) Otherwise, put $d_{k,n}(v) = 0$.

By the properties of $P_{k,n}$, and because $\hat{P}_{k,n}$ is close to $P_{k,n}$, it follows for all n that

$$d_{k,n}(v0) + d_{k,n}(v1) \leq 2d_{k,n}(v)$$

and that

$$\begin{aligned} d_{k,n}(\lambda) &\leq P_{k,n}(\lambda) + \gamma + 2\gamma m \\ &\leq 1/n^3 + \gamma(2m+1) \leq 2/n^3. \end{aligned}$$

Further it follows from the security of G_t that there exist infinitely many n such that for all w having a circuit of size n^k,

$$d_{k,n}(w) \geq \hat{P}_{k,n}(w) \geq 1/2.$$

This means that $d_{k,n}$ is a partial supermartingale which for infinitely many n increases its value by a factor more than n^2 on all w having a circuit of size n^k. It remains to combine the $d_{k,n}$ to some single d_k succeeding on all languages having circuits of size n^k.

Let d_k be defined in the following way. For any language A and a string x of length $n = |x|$, define

$$\begin{aligned} d_k(A[\lambda \ldots x]) = &\sum_{i=0}^{n-1} d_{k,i}(A[x_{i,1} \ldots x_{i,m(i)}]) \\ &+ d_{k,n}(A[x_{n,1} \ldots \min\{x, x_{n,m(n)}\}]) + a_{k,n}, \end{aligned}$$

where $a_{k,n} = \sum_{i=1}^{\infty} 1/i^2 - \sum_{i=0}^{n} d_{k,i}(\lambda)$.

Since for all n, $d_{k,n}$ satisfies the average law and $d_{k,n}(\lambda) \leq 1/n^2$, it follows that d_k is a nonnegative supermartingale.

Furthermore, because for any language A having circuits of size n^k, it holds that for infinitely many n, $d_{k,n}(A[x_{n,1} \ldots x_{n,m(n)}]) \geq 1/2$, d_k succeeds on A.

Finally, observe that d_k on input $A[\lambda \ldots x]$ is computable in time exponential in n (independently of k) relative to some oracle in NP. This completes the proof. □

From Theorem 2 we get the following two sufficient conditions for P/poly having measure 0 at $\mathrm{EXP}^{\mathrm{NP}}$. The first corollary follows immediately from the fact that $\mathrm{EXP}^{\mathrm{NP}}$ is not contained in $\mathrm{P}^{\mathrm{NP}}/\mathrm{poly}$ unless $\mathrm{EXP}^{\mathrm{NP}}$ collapses to $\mathrm{ZPP}^{\Sigma_2^p}$ [BH92, KW95]. The second corollary is obtained by using the assumption that NP does not have measure 0 at EXP to build the pseudorandom generators G_t used in the proof of Theorem 2. (In fact, the assumption that NP does not have measure 0 at EXP implies that there exist generators with subexponential security against subexponential-size oracle circuits that use an NP oracle [NW94, Lut96, AK97].)

Corollary 3. *If* $\mathrm{EXP}^{\mathrm{NP}} \neq \mathrm{ZPP}^{\Sigma_2^p}$ *then* $\mathrm{P}/poly$ *has* $\mathrm{EXP}^{\mathrm{NP}}$*-measure zero.*

Corollary 4. *If* NP *is not small then* $\mathrm{P}/poly$ *has* $\mathrm{EXP}^{\mathrm{NP}}$*-measure zero.*

References

[AK97] V. ARVIND AND J. KÖBLER. On resource-bounded measure and pseudorandomness. In *Proc. 17th Conference on Foundations of Software Technology and Theoretical Computer Science*, Lecture Notes in Computer Science #1346, 235–249. Springer-Verlag, 1997.

[ATZ97] K. AMBOS-SPIES, S. A. TERWIJN, AND X. ZHENG. Resource bounded randomness and weakly complete problems. *Theoretical Computer Science*, **172**(1-2):195–207, 1997.

[BDG90] J. L. BALCÁZAR, J. DÍAZ, AND J. GABARRÓ. *Structural Complexity II*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1990.

[BDG95] J. L. BALCÁZAR, J. DÍAZ, AND J. GABARRÓ. *Structural Complexity I*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, second edition, 1995.

[BFNW93] L. BABAI, L. FORTNOW, N. NISAN, AND A. WIGDERSON. BPP has subexponential time simulations unless EXPTIME has publishable proofs. *Computational Complexity*, **3**:307–318, 1993.

[BH92] H. BUHRMAN AND S. HOMER. Superpolynomial circuits, almost sparse oracles, and the exponential hierarchy. In *Proc. 12th Conference on Foundations of Software Technology and Theoretical Computer Science*, 116–127, 1992.

[BvMR+98] H. BUHRMAN, D. VAN MELKEBEEK, K. REGAN, D. SIVAKUMAR, AND M. STRAUSS. A generalization of resource-bounded measure, with an application. In *Proc. 15th Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science #1373, 161–171. Springer-Verlag, 1998.

[GL89] O. GOLDREICH AND L. A. LEVIN. A hard-core predicate for all one-way functions. In *Proc. 21st ACM Symposium on Theory of Computing*. ACM Press, 1989.

[GNW95] O. GOLDREICH, N. NISAN, AND A. WIGDERSON. On Yao's XOR-lemma. Technical report, ECCC, 1995.

[Kan82] R. KANNAN. Circuit-size lower bounds and nonreducibility to sparse sets. *Information and Control*, **55**:40–56, 1982.

[KL80] R. M. KARP AND R. J. LIPTON. Some connections between nonuniform and uniform complexity classes. In *Proc. 12th ACM Symposium on Theory of Computing*, 302–309. ACM Press, 1980.

[KW95] J. KÖBLER AND O. WATANABE. New collapse consequences of NP having small circuits. In *Proc. 22nd International Colloquium on Automata, Languages, and Programming*, Lecture Notes in Computer Science #944, 196–207. Springer-Verlag, 1995.

[Lip91] R. J. LIPTON. New directions in testing. In J. Feigenbaum and M. Merritt, editors, *Distributed Computing and Cryptography*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science #2. American Mathematical Society, 1991.

[Lub97] M. LUBY. *Pseudorandomness and Cryptographic Applications*. Princeton University Press, 1997.

[Lut92] J. H. LUTZ. Almost everywhere high nonuniform complexity. *Journal of Computer and System Sciences*, **44**:220–258, 1992.

[Lut96] J. H. LUTZ. Observations on measure and lowness for Δ_2^{P}. In *Proc. 13th Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science #1046, 87–97. Springer-Verlag, 1996.

[May94] E. MAYORDOMO. *Contributions to the Study of Resource Bounded Measure*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, 1994.

[NW94] N. NISAN AND A. WIGDERSON. Hardness vs randomness. *Journal of Computer and System Sciences*, **49**:149–167, 1994.

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

[RSC95] K.W. REGAN, D. SIVAKUMAR, AND JIN-YI CAI. Pseudorandom generators, measure theory, and natural proofs. In *Proc. 36th IEEE Symposium on the Foundations of Computer Science*. IEEE Computer Society Press, 1995.

[Sto85] L. STOCKMEYER. On approximation algorithms for #P. *SIAM Journal on Computing*, **14**(4):849–861, 1985.

[Yao82] A. C. YAO. Theory and applications of trapdoor functions. In *Proc. 23rd IEEE Symposium on the Foundations of Computer Science*, 80–91. IEEE Computer Society Press, 1982.