Pseudorandom Generators, Measure Theory, and Natural Proofs

KENNETH W. REGAN D. SIVAKUMAR JIN-YI CAI

Department of Computer Science,
State University of New York at Buffalo,
Buffalo, NY 14260.
Email: {regan, sivak-d, cai}@cs.buffalo.edu*

Abstract

This paper proves that if strong pseudorandom number generators or one-way functions exist, then the class of languages that have polynomial-sized circuits is not small within exponential time, in terms of the resource-bounded measure theory of Lutz. More precisely, if for some $\varepsilon > 0$ there exist nonuniformly 2^{n^ε}-hard PSRGs, as is widely believed, then P/poly does not have measure zero in EXP. Our results establish connections between the measure theory and the "natural proofs" of Razborov and Rudich. Our work is also motivated by Lutz's hypothesis that NP does not have measure zero in EXP; obtaining our results with NP in place of P/poly would show much more far-reaching consequences from the existence of PSRGs than are currently known.

1. Introduction

The theory of resource-bounded measure, initiated by Lutz [14, 15, 16] and furthered in [20, 12, 17, 18, 19, 13, 1, 21], has provided a useful framework that links many central problems in complexity theory. Classes that have measure zero are *small* in a quantitative sense described by Lutz in [16]. Lutz et al. [16, 12, 18] have advanced the hypothesis that NP does not have measure zero within EXP, where EXP stands for DTIME$[2^{n^{O(1)}}]$, and have shown that several striking and plausible consequences would follow: Besides NP $\neq$ P, there would be NP-complete languages under polynomial-time Turing reductions that are not complete under many-one reductions, NP would contain immune and bi-immune sets for P, and there would be NP search problems that do not reduce to their corresponding decision problems.

We first prove that if NP $\neq$ EXP, then either NP has measure zero or NP is not measurable at all within EXP. Hence Lutz's conjecture is really that NP is not measurable. Up to now there have not been any general techniques for showing classes $\mathcal{C}$ to be *non*-measurable. Our paper gives such a technique, based on the theory of pseudorandom generators (PSRGs) ([5, 10, 8, 9]). PSRGs that have *exponential hardness*, meaning that for some $\varepsilon > 0$ they are unbreakable by 2^{n^ε}-sized circuits, are widely believed to exist. Indeed, the smallest circuits known to break PSRGs based on the discrete logarithm problem have size just short of $2^{n^{1/2}}$. Our main theorem, however, is for $\mathcal{C}$ = P/poly, the class of all languages having polynomial-sized circuits, rather than $\mathcal{C}$ = NP.

*Regan and Sivakumar were supported in part by NSF Grant CCR 9409104. Cai was supported in part by NSF Grants CCR-9057486 and CCR-9319093, and by an Alfred P. Sloan Fellowship.

Theorem 1. *If there exist PSRGs of exponential hardness or one-way functions of exponential security against non-uniform adversaries, then* P/poly *is not measurable in* EXP*.*

This is interesting because ordinarily one thinks of P/poly as a tractable class, much lower down than NP on the complexity scale. What our proof brings out is the large role played by *nonuniformity*. It also follows from our results that if there is a *pseudorandom function generator* [7] of exponential security in nonuniform NC^1, then nonuniform NC^1 is not measurable in EXP. We also prove unconditionally that non-uniform $AC^0[2]$ is not measurable under either of the measures defined by Allender and Strauss [2].

We prove our theorems by showing that the *martingales* used in defining Lutz's measure theory yield *natural properties*, as defined by Razborov and Rudich [26], of equivalent non-uniform complexity that diagonalize against (or "are useful against") the class $\mathcal{C}$ on which the martingale succeeds. An improvement of theorems in [26] due to Razborov [25] yields the conclusion. The main technical problem solved in our proof is that the martingales are defined on "characteristic prefixes" that define membership in a language L of all strings *up to* a given length n, whereas the natural properties concern length-n only. Our solution appears to work only for non-uniform complexity.

The natural properties produced by the martingales have greater density—$1/n^{O(1)}$ in place of $1/2^{O(n)}$—than that postulated by Razborov and Rudich. We note that the first four main examples in [26] actually have density at least constant or $1 - o(1)$, and *all* of them meet a stronger "a.e." condition of diagonalizing against every $L \in \mathcal{C}$ at all but finitely many lengths, not just infinitely many lengths. In addressing the question of whether the measure and natural-proof theories are *equivalent*, we prove the following:

Theorem 2.

(a) For every $\varepsilon > 0$ and P/poly*-computable martingale that succeeds on* P/poly*, there is a* P/poly*-natural property of density $1/n^{1+\varepsilon}$ that diagonalizes* i.o. *against* P/poly*.*

(b) For every P/poly*-natural property of density $1/n$ that diagonalizes* a.e. *against* P/poly*, there is a* P/poly*-computable martingale that succeeds on* P/poly*.*

These relationships hold for other well-behaved classes besides P/poly. Our results give strong reasons to investigate further both the measure theory and the natural-proofs theory, promising progress on important problems in complexity theory.

2. Preliminaries

The notation and conventions we use are essentially standard. All languages and functions are assumed to be defined over the finite alphabet $\Sigma = \{0, 1\}$. The empty string is denoted by λ. We denote by F_n the set of all Boolean functions in n variables. A Boolean function $f_n \in F_n$ can be thought of as a binary string of length 2^n that represents the *truth table* of f_n. For readability we often write N for 2^n. We identify a language A with its characteristic sequence χ_A, and regard the latter also as a member of the set $\{0,1\}^\omega$ of infinite binary strings. For all $n \geq 0$ we also identify $A^{=n}$ with the segment u_n of χ_A of length 2^n that represents the membership or nonmembership in A of all strings of length n, and likewise identify $A^{\leq n}$ with $u_0 u_1 \cdots u_n$. Note that each u_n belongs to F_n. Then the *cylinder* $C_w = \{z \in \{0,1\}^\omega : w \sqsubseteq z\}$ contains A and all languages that agree with A on the membership of strings up to the last one indexed by w, under the standard ordering of Σ^*.

QP stands for DTIME$[2^{\mathrm{polylog}\, n}]$, which is often called *quasipolynomial time*. QP/qpoly stands for the class of languages accepted by quasipolynomial-sized circuits; equivalently, by $2^{\mathrm{polylog}\, n}$-time bounded Turing machines that take $2^{\mathrm{polylog}\, n}$ bits of *advice*. This is analogous to P/poly but for quasipolynomial bounds. All logarithms in this paper are to the base 2.

2.1. Pseudorandom generators and one-way functions

A PSRG is formally a sequence $\{G_n\}$, where each G_n is a function from $\{0,1\}^n$ to $\{0,1\}^{\ell(n)}$, and $\ell(n) > n$. Intuitively, G_n is designed to "stretch" a sequence of n truly random bits into a longer sequence of bits that appear random to resource-bounded adversaries.

Definition 1. For a PSRG $G = \{G_n\}$, its *hardness at n, $H(G_n)$,* is defined to be the largest integer $S(n)$ such that for every $\ell(n)$-input circuit C of size at most $S(n)$,

$$\left| \Pr_{y \in \{0,1\}^{\ell(n)}}[C(y) = 1] - \Pr_{x \in \{0,1\}^n}[C(G_n(x)) = 1] \right| \leq \frac{1}{S(n)}.$$

G is said to be of *hardness at least* $h(\cdot)$ if for all but finitely many n, $H(G_n) \geq h(n)$.

A well-known "robustness" theorem (see [5, 9]) states that so long as $\ell(n) = n^{O(1)}$, $H(G_n)$ is invariant up to constant factors. As Razborov and Rudich do, we work with PSRGs that stretch n bits to $2n$ bits.

We mention in-passing the results of Hastad, Impagliazzo, Levin, and Luby [9, 10, 8] proving that for any resource-bound class $\mathcal{R}$ between $n^{O(1)}$ and $2^{n^{o(1)}}$ that is invariant under polynomial scaling, there exists a PSRG of hardness greater than $\mathcal{R}$ iff there exists a *one-way function* that is *secure* against $\mathcal{R}$-bounded adversaries. The equivalence holds also in the uniform case [8, 9], where the adversaries are time-bounded probabilistic Turing machines (PTMs) and $\mathcal{R}$ bounds running time rather than circuit size. The transformations in [9] yield the following, which we cite for later reference.

Theorem 3. ([9]) *If for some constant $\gamma > 0$, there exists a one-way function of security 2^{n^γ} against non-uniform (resp. uniform) adversaries, then for some constant $\delta > 0$ there exists a pseudorandom generator of hardness 2^{n^δ} against non-uniform (resp. uniform) adversaries.*

2.2. Resource-bounded measure

The resource-bounded measure theory of Lutz [15, 16] is developed along the lines of classical measure theory (see [23, 6, 24]). Languages are regarded as points in the topological space whose basic open sets are the "cylinders" C_w, one for each $w \in \{0,1\}^*$, and complexity classes are point sets. The general form of Lutz's theory, expounded recently by Mayordomo [21], defines conditions for a class $\mathcal{C}$ to be *measurable* by a function class Δ, and to have *measure* e, written $\mu_\Delta(\mathcal{C}) = e$, where $0 \leq e \leq 1$. Since all complexity classes we discuss are closed under finite variations, and by a form of the *Kolmogorov zero-one law* proved in [21] have measure zero or one, we need only discuss conditions for classes to have measure zero. This thesis and [15, 16] show that these measurability conditions can be defined in terms of *martingales* of the kind studied earlier by Schnorr [29, 30, 31]. A *martingale* is a function d from $\{0,1\}^*$ into the nonnegative reals that satisfies the following "exact average law": for all $w \in \{0,1\}^*$,

$$d(w) = \frac{d(w0) + d(w1)}{2}. \tag{1}$$

Let $\mathbf{D}$ stand for the nonnegative dyadic rationals; i.e., those numbers of the form $n/2^r$ for integers $n, r \geq 0$.

Definition 2 (compare [15, 21]). Let Δ be a complexity class of functions. A class $\mathcal{C}$ of languages is Δ-*measurable and has* Δ-*measure zero*, written $\mu_\Delta(\mathcal{C}) = 0$, if there is a martingale $d : \{0,1\}^* \to \mathbf{D}$ computable in Δ that *succeeds on* $\mathcal{C}$, in the sense that $\mathcal{C} \subseteq S^\infty[d]$ where

$$S^\infty[d] = \{A : \lim_{w \sqsubseteq A} d(w) = +\infty\}.$$

Put another way, the *success class* $S^\infty[d]$ is the class of languages A that satisfy

$$(\forall K > 0)(\exists N > 0)(\forall w \sqsubseteq A)[|w| \geq N \Rightarrow d(w) \geq K]. \tag{2}$$

Intuitively, the martingale d is a "betting strategy" that starts with a capital sum $d(\lambda) > 0$ and makes infinite profit along the characteristic strings of every $A \in S^\infty[d]$. The purpose of the theory is to analyze the complexity required for a martingale to succeed on every language in certain subclasses $\mathcal{C}$ of a given class $\mathcal{D}$. This provides a tool for analyzing the internal structure of $\mathcal{D}$.

If $\mathcal{D}$ is defined by a collection $\mathcal{R}$ of resource bounds that is closed under squaring, then Lutz defines $\Delta(\mathcal{D})$ to be the class of martingales computable within the bound $r(\log N)$ for some function $r(N) \in \mathcal{R}$. For any class $\mathcal{C}$, Lutz writes $\mu(\mathcal{C}|\mathcal{D}) = 0$, read "$\mathcal{C}$ has measure zero within $\mathcal{D}$," if $\mu_{\Delta(\mathcal{D})}(\mathcal{C} \cap \mathcal{D}) = 0$. Two instances of particular importance are:

$$\begin{array}{ll} \mathcal{D} = \mathrm{E}, & \Delta = \mathrm{P}, \\ \mathcal{D} = \mathrm{EXP}, & \Delta = \mathrm{QP}. \end{array}$$

Mayordomo [21] proved that in these cases, the definition of $\mu(\mathcal{C}|\mathcal{D}) = 0$ is robust under certain changes to Definition 2, most notably under relaxing (1) to the inequality $d(w) \geq (d(w0) + d(w1))/2$, and under relaxing the limit condition in (2) to a limsup; viz. for all $A \in \mathcal{C}$, $(\forall K > 0)(\exists w \sqsubseteq A)[d(w) \geq K]$. Hence the above is equivalent to the formulations originally used by Lutz in [15]. (We return to the robustness issue when discussing the measures on $\mathcal{D} = \mathrm{P}$ defined by Allender and Strauss [2].)

If $\mu(\mathcal{C}|\mathcal{D}) = 0$, then $\mathcal{C} \cap \mathcal{D}$ is intuitively "small" as a subclass of $\mathcal{D}$. If NP is small within E [or within EXP], then there is a single [quasi-]polynomial time computable betting strategy that succeeds simultaneously against every language in NP. With the feeling that this is unlikely, Lutz advanced the conjecture that NP is not small in either class. The striking consequences listed in the Introduction follow if NP is not small (in particular, the consequences are shown in [12, 16, 18] to follow from $\neg\mu_{\mathrm{P}}(\mathrm{NP}) = 0$, which is implied by $\neg\mu_{\mathrm{QP}}(\mathrm{NP}) = 0$).

The classical time-hierarchy theorems carry over to measure; in particular, P and QP have measure zero in E, and E itself, indeed DTIME$[2^{n^c}]$ for any fixed c, has measure zero in EXP. It is shown in [15, 16, 2] that classes of measure zero behave very much like null-sets in classical measure theory. The complement (in $\mathcal{D}$) of a measure-zero subclass $\mathcal{C}$ has $\Delta(\mathcal{D})$ measure 1 (this is a definition in [15, 16] and a theorem in [21]). Finite unions, and also "$\Delta(\mathcal{D})$-bounded" countable unions, of measure-zero classes have measure zero. The operation of symmetric difference on languages, namely $A \triangle B = (A \setminus B) \cup (B \setminus A)$, behaves like an "affine translation" in preserving measure:

Proposition 4. *Let $\mathcal{D}$ be a time complexity class for which $\Delta(\mathcal{D})$ is definable as above, and given $\mathcal{C} \subseteq \mathcal{D}$ and $A \in \mathcal{D}$, define $\mathcal{C} \triangle A = \{L \triangle A : L \in \mathcal{C}\}$. Then $\mu(\mathcal{C}|\mathcal{D}) = 0 \iff \mu(\mathcal{C} \triangle A|\mathcal{D}) = 0$.*

Proof. As a function of $N = |w|$, a $\Delta(\mathcal{D})$ machine M has enough time to simulate a fixed $\mathcal{D}$-machine M_A that accepts A on all the strings indexed by bits in w, forming the length-N initial segment v of χ_A. Then letting d be the original martingale that succeeds on $\mathcal{C}$, M outputs $d(w \oplus v)$. □

(This result holds also for the Allender-Strauss measures, since M_A need be simulated only on those x indexed by bits in the "dependency set" for $d(w)$.)

Lutz [16] mentions that his hypothesis that NP is not small in EXP leaves open the possibility that NP has measure 1 in EXP, or that NP is not μ_{QP}-measurable at all. The following new result essentially removes one of these possibilities.

Theorem 5. *With $\mathcal{D}$ as above, let $\mathcal{C}$ be a proper subclass of $\mathcal{D}$ that is closed under symmetric difference, or under finite union and intersection. Then $\mathcal{C}$ does not have measure 1 in $\mathcal{D}$.*

Proof. If $\mathcal{C}$ has measure 1, then $\mathcal{D} \setminus \mathcal{C}$ has measure zero. Because $\mathcal{D}$ is a deterministic time class, it is closed under all Boolean operations, and it follows that $(\mathcal{D} \setminus \mathcal{C}) \triangle \Sigma^* = \mathcal{D} \setminus$ co-$\mathcal{C}$. Hence by Proposition 4, co-$\mathcal{C}$ has measure 1 in $\mathcal{D}$. So does $\mathcal{C}' = \mathcal{C} \cap$co-$\mathcal{C}$, since the intersection of two measure-1 subclasses of $\mathcal{D}$ has measure 1. Now $\mathcal{C}'$ is closed under symmetric difference, so if we let $A \in \mathcal{D} \setminus \mathcal{C}'$, $\mathcal{C}' \triangle A$ is disjoint from $\mathcal{C}'$. But $\mathcal{D}$ cannot contain two disjoint measure-1 subclasses. □

Corollary 6. *Let $\mathcal{C}$ denote any of* NP, coNP, Σ_k^p, Π_k^p, P/poly, *nonuniform* NC, BPP, PP, *or* PSPACE. *Then $\mu(\mathcal{C}|\mathrm{EXP}) = 1 \iff \mathcal{C} = \mathrm{EXP} \iff \mathcal{C} \cap$ co-$\mathcal{C} = \mathrm{EXP}$.*

2.3. Natural Proofs

The technical concept at the heart of the paper by Razborov and Rudich [26] is the following. For each n, let F_n denote the set of n-variable Boolean functions; i.e., functions from $\{0,1\}^n$ to $\{0,1\}$. Then $\|F_n\| = 2^N$, where $N = 2^n$. Define a *combinatorial property* to be a sequence $\Pi = [\Pi_n]_{n=0}^{\infty}$, where each Π_n is a subset of F_n. A language A is *drawn from* Π if for all n, the Boolean function given by $A^{=n}$ belongs to Π_n. The property Π *diagonalizes over* a class $\mathcal{C}$ of languages, or "is useful against" $\mathcal{C}$, if no language drawn from Π belongs to $\mathcal{C}$. When $\mathcal{C}$ is closed under finite variations, this is equivalent to diagonalizing *i.o.* against $\mathcal{C}$:

$$(\forall B \in \mathcal{C})(\exists^{\infty} n)\ B^{=n} \notin \Pi_n. \tag{3}$$

We remark that all of the natural properties constructed in [26] satisfy the stronger condition

$$(\forall B \in \mathcal{C})(\forall^{\infty} n)\ B^{=n} \notin \Pi_n. \tag{4}$$

We call this *diagonalizing a.e. against* $\mathcal{C}$. Indeed, the journal version [27] of their paper adopts the "a.e." definition, by inserting a clause "for all sufficiently large n" into the conference version's definition of "useful." Both the "i.o." and the "a.e." conditions are important in this paper, and our work below on going from natural proofs to martingales brings out the significance of the difference.

The complexity of Π is the complexity of the decision problem: given a Boolean function $f_n \in F_n$, is $f_n \in \Pi_n$? Note that this complexity is measured as a function of N, not of n, so it has the same "scaling" as Lutz's martingales. Finally, the property is *large* if there exists a polynomial p such that for all but finitely many n,

$$\rho(\Pi_n) = \frac{\|\Pi_n\|}{2^N} \geq \frac{1}{p(N)}. \tag{5}$$

Put another way, the Boolean functions in Π_n have *non-negligible* density in the space of all Boolean functions.

Definition 3 (cf. [26]). Let $\mathcal{C}$ and $\mathcal{D}$ be complexity classes of languages. A combinatorial property Π is $\mathcal{D}$*-natural against* $\mathcal{C}$ if Π is large, belongs to $\mathcal{D}$, and diagonalizes over $\mathcal{C}$.

Rudich and Razborov show that several important separation results in complexity use techniques that construct natural properties. Two of their main theorems point out limitations of such techniques. The following two improvements of these theorems from polynomial to quasipolynomial size bounds for $\mathcal{D}$ are noted by Razborov [25]:

Theorem 7.

(a) If there exists a combinatorial property that is $\mathrm{QP/qpoly}$*-natural against* $\mathrm{P/poly}$*, then PSRGs of exponential hardness against non-uniform adversaries do not exist.*

(b) There does not exist a combinatorial property that is $q\mathrm{AC}^0$*-natural against* $\mathrm{AC}^0[2]$*, where* $q\mathrm{AC}^0$ *denotes the class of languages accepted by a quasipolynomial size circuit family of constant depth.*

(In **Appendix 1**, we sketch the needed changes to the proofs in [26], which are not given by Razborov in [25], nor in [27]. These were discovered independently, but later, by us.)

3. Main Results

To prove our main theorem, we show that if $\mu(\mathrm{P/poly}|\mathrm{EXP}) = 0$, then one can build a natural property that diagonalizes over P/poly. Since the measure on EXP involves QP-computable martingales, we obtain a natural property that belongs to QP/qpoly, in fact to quasipolynomial time with linear (in N) advice. Our first lemma follows by an elementary counting argument, using the fact that $\sum_{v\in\{0,1\}^\ell} d(uv) = 2^\ell \cdot d(u)$.

Lemma 8. *Let d be a martingale. For any string u and any $\ell \in \mathbf{N}$, $b \in \mathbf{R}$,*

$$\left\| \{v \in \{0,1\}^\ell : d(uv) \leq \left(1+\frac{1}{b}\right) d(u)\} \right\| \geq 2^\ell \left(\frac{1}{b+1}\right).$$

Our key lemma has the idea that given a martingale d that succeeds on P/poly, we can build a combinatorial property that captures those Boolean functions on $\{0,1\}^n$ along which d makes too little income to succeed. This property then diagonalizes i.o. against the success class of the martingale, which contains P/poly. Since $\Pi_n(1+1/n^2)$ converges, we can say that a return on capital of $1/n^2$, let alone losing money along a branch, is "too little income" for d. Lemma 8 will guarantee that the density of these poor branches is at least $1/n^2 = 1/(\log^2 N)$, a notably greater density than that called "large" in [26].

Lemma 9. *If a* QP *martingale d succeeds on* P/poly $\cap$ EXP *then for every polynomial q, there exist infinitely many n and circuits C_i of size at most $q(i)$, for $0 \leq i < n$, such that for all circuits C_n of size at most $q(n)$,*

$$d(u_0 \ldots u_n) > \left(1+\frac{1}{n^2}\right) d(u_0 \ldots u_{n-1}),$$

where u_i is the 2^i-bit binary "characteristic string" that indicates the membership in $L(C_i)$ of $\{0,1\}^i$.

Proof. Suppose not. Then there is a polynomial q and constant $n_0 \in \mathbf{N}$ such that for all $n \geq n_0$, for every sequence of circuits C_i of size at most $q(i)$, for $0 \leq i < n$, there exists a circuit C_n of size at most $q(n)$ such that $d(u_0 \ldots u_n) \leq \left(1+\frac{1}{n^2}\right) d(u_0 \ldots u_{n-1})$, where the u_i's have the same meaning as in the statement of the lemma.

We will build a language L as follows: for strings of length less than n_0, membership in L will be an arbitrary but fixed sequence. Let $\alpha = d(u_0 \ldots u_{n_0-1})$. Clearly $\alpha < \infty$. For $n \geq n_0$, we define $L^{=n}$ inductively. Let $u_0, \ldots, u_{n-1}$ be the result of the recursively applying the construction to obtain $L^{<n}$; that is, $u_i = L^{=i}$. By assumption, there exists a circuit C_n of size at most $q(n)$ such that $d(u_0 \ldots u_n) \leq \left(1+\frac{1}{n^2}\right) d(u_0 \ldots u_{n-1})$. Set $u_n = L(C^*)^{=n}$, where C^* is the lexicographically first C_n that satisfies this inequality (under some fixed encoding of circuits of size at most $q(n)$).

Clearly $L \in$ P/poly, since it can be accepted by the circuit family $[C_n]_{n=0}^\infty$. That $L \in$ EXP is immediate from the fact that finding the lexicographically first C_n takes time at most $2^{q(n)+p(n)}$, where the running time to compute the martingale d determines $p(n)$. Finally,

$$\lim_{n\to\infty} d(L^{\leq n}) \leq \alpha \prod (1 + 1/n^2) < \infty,$$

so d does not succeed on L, a contradiction. $\square$

The remaining technical problem is to weave together the constructions in Lemma 9 for all polynomial bounds q. We do not know of a *uniform* way to choose the circuits $C_0, C_1, \ldots, C_{n-1}$ promised by Lemma 9 over all q and the infinitely-many n for each q, and this is where nonuniformity enters into our results.

Lemma 10. *If* $\mu(\mathrm{P/poly}|\mathrm{EXP}) = 0$*, then there is a* QP/poly*-natural property against* P/poly.

Proof. For each k, let T_k be the infinite set of numbers n promised by Lemma 9 for the bound $q(n) = n^k$. Set $T := \cup_k T_k$. For all $n \in T$, take the largest number $k \leq n$ such that $n \in T_k$, take the lexicographically first $C_0, \ldots, C_{n-1}$ that works in Lemma 9, and define U_{n-1} to be the concatenation of the corresponding $u_0, \ldots, u_{n-1}$. For $n \notin T$, make some arbitrary choice such as $U_{n-1} = 0^{2^n - 1}$. Finally, for all n define

$$\Pi_n := \left\{ f_n : d(U_{n-1} f_n) \leq \left(1 + \frac{1}{n^2}\right) d(U_{n-1}) \right\}.$$

Now, by Lemma 8, the property $\Pi = \{\Pi_n\}$ is large; in fact, it has density $1/\operatorname{poly}(n)$, not just $1/\operatorname{poly}(2^n)$. By the computability of the martingale d, Π_n can be recognized in quasi-polynomial time in 2^n, given the U_{n-1}'s as advice. Equivalently, there is a family of circuits of size quasi-polynomial in 2^n that recognizes Π_n. Let L be an arbitrary language in P/poly, and let n^k be a bound on the size of a family of circuits to recognize L. Clearly, for all $n \in T_k$, $L^{=n} \notin \Pi_n$. Therefore, property Π diagonalizes i.o. over P/poly. $\square$

Theorem 11. *If* $\mu(\mathrm{P/poly}|\mathrm{EXP}) = 0$*, then for every family of pseudorandom generators* $G = \{G_k : \{0,1\}^k \to \{0,1\}^{2k}\}$ *computable in* P/poly*, for every* $\varepsilon > 0$*, for sufficiently large values of* k*,* $H(G_k) \leq 2^{k^\varepsilon}$.

Proof. This follows from the above three lemmas and Theorem 7. $\square$

Corollary 12. *If for some* $\gamma > 0$ *there exists a one-way function of security* 2^{n^γ}*, then* P/poly *is not measurable in* EXP. $\square$

Based on assumptions about the hardness of the *subset-sum* problem, Impagliazzo and Naor [11] show how to construct a pseudorandom generator in NC^1. Razborov and Rudich note that if there is a pseudorandom *function* generator of exponential hardness in NC^1, there is no P/poly-natural proof against NC^1. It follows from our results that:

Theorem 13. *If there is a pseudorandom function generator of exponential hardness in* NC^1*, then nonuniform* NC^1 *is not measurable in* EXP. $\square$

3.1. Measure of $\mathrm{AC}^0[2]$

Allender and Strauss [2] define measures on the class $\mathcal{D} = \mathrm{P}$, imposing a restriction on the corresponding martingale class that becomes vacuous for $\mathcal{D} = \mathrm{E}$ or $\mathcal{D} = \mathrm{EXP}$, and can be described as follows: Rather than give the Turing machines M computing martingale values $d(w)$ the string w as input, give them $N = |w|$ in binary notation on their input tape, and let them query individual bits of w. (Then M is formally the same as the machines used to define the PCP classes in [4, 3, 32].) Measure time bounds in terms of $n = \lceil \log_2 N \rceil = |N|$ rather than N. Then the function $d(\cdot)$ belongs to $\Gamma(\mathrm{P})$ as defined in [2] if M runs in time $n^{O(1)}$, and if every node N in the directed "dependency graph," defined to have an edge (m, N) if M on input N queries bit m of *some* w, has $n^{O(1)}$ predecessors. They write $\mu(\mathcal{C}|\mathrm{P}) = 0$ if there is a $\Gamma(\mathrm{P})$ martingale that succeeds on $\mathcal{C} \cap \mathrm{P}$.

Allender and Strauss note that their measure is robust under either one of the relaxations mentioned in section 2.2, but that relaxing *both*, i.e. allowing $d(w) \geq (d(w0) + d(w1))/2$ in place of (1) and using the "limsup" condition of success in place of (2), yields a different measure. We write $\mu_2(\mathcal{C}|\mathrm{P}) = 0$ to signify that $\mathcal{C}$ is one of the strictly-larger family of null classes in their second measure. They show that the class of sparse sets in P is null in the latter but not the former, and in particular that (P-uniform) AC^0 is not $\Gamma(\mathrm{P})$-measurable. But whether $\mu_2(\mathrm{AC}^0|\mathrm{P}) = 0$ is open. Using our methods, we show:

Theorem 14. *Nonuniform* $\mathrm{AC}^0[2]$ *does not have* μ_2 *measure zero.*

Proof Sketch. The main idea is that owing to the dependency-set restriction in defining $\Gamma(\mathrm{P})$, the hypothesis $\mu_2(\mathrm{AC}^0[2]) = 0$ yields a qAC^0-natural property against $\mathrm{AC}^0[2]$. To handle the fact that the notion of $\Gamma_2(\mathrm{P})$ measure is defined using lim sup rather than the limit, we use stronger versions of Lemmas 8 and 9. Theorem 7(b) then yields a contradiction. $\square$

(**Appendix 2** contains a longer, more-detailed discussion of the Allender-Strauss measures and a full proof of Theorem 14. It also remarks on the problem of strengthening this to read: nonuniform $\mathrm{AC}^0[2]$ is not measurable *in* P.)

4. The Uniform Case and Honest Martingales

The next interesting question is whether Theorem 11 can be made to work under the hypothesis that for some $\gamma > 0$ there is a one-way function of security 2^{n^γ} *against uniform adversaries*. The main problem is that the natural property we construct in Proposition 10 is non-uniform, and this non-uniformity carries over to the statistical test constructed in the theorem of Razborov and Rudich, drawing on [7]. That is, the property belongs to QP/poly. We have not been able to obtain a QP-natural property under the hypothesis $\mu(\mathrm{P/poly}|\mathrm{EXP}) = 0$—the sticking point is that we have not been able to enforce any "consistency" among the characteristic prefixes $u_0, \ldots, u_{n-1}$ obtained in applications of Lemma 9 to build the Π_k that are interleaved in the proof of Lemma 10.

Interest in this problem led us to define the following "prefix-invariance" restriction on martingales, which also comes up naturally in the next section. We begin by formalizing the associated concept of a *betting strategy*.

Definition 4. A *betting strategy* is any function $b : \{0,1\}^* \to [-1 \ldots +1]$. The *martingale* d_b *derived from* b is defined by $d_b(\lambda) = 1$, and for all $w \in \{0,1\}^*$, $d_b(w1) = d_b(w)(1 + b(w))$, $d_b(w0) = d_b(w)(1 - b(w))$.

For all w, let x_w stand for the string indexed by the bit c in wc, and let n_w be the length of x_w; i.e., $n_w = \lfloor \log_2(|w| + 1) \rfloor$. Intuitively, $b(w)$ is the signed proportion of current capital bet on the event that x_w belongs to a given language L. A negative value of $b(w)$ indicates a bet that $x_w \notin L$. Given a martingale d, one can regard the function $b_d(w) := (d(w1) - d(w))/d(w)$ as the associated betting strategy, although it is undefined when $d(w) = 0$. The possibility that $d(w) = 0$ for some w is actually ruled out by the condition that the "success class" of d is closed under finite variations (or, it can be avoided by a straightforward modification of d to have all values $\geq \frac{1}{2}d(\lambda)$), but we have no difficulty with this possibility being left open.

Definition 5. A martingale $d : \{0,1\}^* \to \mathbf{R}$ is *honest* if it is derived from a betting strategy $b : \{0,1\}^* \to \mathbf{R}$, such that for all $w \in \{0,1\}^*$, the computation of $b(w)$ depends only on those parts of w that index strings of length n_w.

Many of the martingales implicitly constructed by Lutz et al. are honest, [1] and this condition deserves further investigation. For honest martingales we immediately obtain a stronger form of Lemma 9:

Lemma 15. *If an honest* QP *martingale* d *succeeds on* P/poly $\cap$ EXP *then for every polynomial* q*, there exist infinitely many* n *such that for all circuits* C_n *of size at most* $q(n)$*, and all characteristic prefix strings* $w \in \{0,1\}^{2^n - 1}$*,*

$$d(wu_n) \geq \left(1 + \frac{1}{n^2}\right) d(w),$$

where u_n *is the binary characteristic string of length* 2^n *that represents the strings accepted and rejected by* C_n*.* □

Theorem 16. *If a honest martingale computable in quasi-polynomial time succeeds on* P/poly $\cap$ EXP*, then one-way functions (and pseudorandom generators) of security* 2^{k^γ} *against uniform adversaries do not exist.*

Proof. Let the honest QP-computable martingale d be given. For all n, let w_n be some characteristic prefix of length $N - 1$ (indexing strings of lengths 0 through $n - 1$) such that

[1] Some exceptions are the "incompressibility theorem" of Juedes and Lutz [12], and theorems that build martingales that "look back" in the input string for specific properties.

$d(w_n) > 0$. Such w_n can be found in quasipolynomial time by starting at the root λ and always taking branches along which $d(\cdot)$ does not decrease. (Or we can assume as remarked above that d takes nonzero values and use $w_n = 0^{N-1}$.) For all n, define

$$\Pi_n = \{u \in \{0,1\}^N : d(w_n u) \leq (1 + \frac{1}{n^2})d(w_n)\}.$$

The corresponding property $\Pi = \{\Pi_n\}$ is large and belongs to QP. By Lemma 15, and with the step of fixing "$u_0, \ldots, u_{n-1}$" in the proof of Lemma 10 now rendered unnecessary, it follows that Π diagonalizes i.o. over P/poly.

It remains to verify that the statistical test constructed from Π by Razborov and Rudich, drawing on [7], is computable by a probabilistic Turing machine in time less than 2^{k^γ}. Let $d(w)$ be computable in time $2^{(\log |w|)^c}$ on inputs w. Let a PSRG $G : \{0,1\}^k \to \{0,1\}2k$ be given. Given γ, take $\varepsilon < \gamma$, and take $n = k^{\varepsilon/c}$. (In [26], with a P/poly-natural property, they have $c = 1$.) Recall the construction of "$G_y(x)$" in [26, 7], and of the pseudorandom function generator defined by $f_x(y)$ = the first bit of $G_y(x)$. Here $|x| = k$ and $|y| = n$, and x can be thought of as "advice" to compute the pseudorandom function f in time polynomial in n.

Now by the proofs of Lemmas 9 and 10, for every polynomial $q(n)$, there are infinitely many n such that Π_n diagonalizes over all languages acceptable in time and advice $q(n)$; we take $q(n) = n^{c/\varepsilon}$. Then for $k = q(n)$, Π_n can be used as a statistical test against f, along the lines laid out by Razborov and Rudich, using the tree-construction from [7]. The only nonuniform step in their proof is the fixing of strings x_v for all roots v of subtrees that come before the subtree being isolated (v_{i+1} in the proof, with respect to the ordering used there), via an "averaging argument." When we have a probabilistic Turing machine M, however, the averaging argument can be dispensed with: M flips coins to select i, $1 \leq i \leq 2^n$, locates v_i and v_{i+1}, and then randomly assigns k-bit strings x_{v_j} to all nodes v_j with $j < i$. Then the 2^n leaves of the tree yield a bit-string of length N that is tested for membership in Π_n. The verification that M obeys the required time bounds and achieves the necessary bias is placed into Appendix 1. $\square$

A slightly stronger result follows from the above: if there is a uniform P-natural or even QP-natural proof against P/poly $\cap$ EXP, not just against P/poly, then there are no PSRGs of hardness 2^{n^γ} against uniform adversaries. This leads into a *very sensitive* point about the interplay between uniformity and non-uniformity, deserving its own subsection.

4.1. Uniform and non-uniform results

The above results hold in a fairly general form:

Theorem 17. *Let $\mathcal{D}$ be a uniform complexity class defined by a collection $\mathcal{R}$ of time bounds on Turing machines that contains $O(N)$ and is closed under squaring, with nonuniform-$\mathcal{D}$ defined by bounds in $\mathcal{R}$ on circuit size. Then:*

(a) *For every martingale d computable in non-uniform $\mathcal{D}$, we can construct a natural property Π belonging to non-uniform $\mathcal{D}$ such that Π has density at least $1/\log^2(N)$ and diagonalizes i.o. against the success class $S^\infty[d]$ of the martingale.*

(b) *If d is in $\mathcal{D}$,* and *if d is honest, then Π belongs to $\mathcal{D}$; i.e., Π is uniform.*

Similar results apply, with technical modifications, for the polylog(N)-time $\Gamma_2(\mathrm{P})$ measure of Allender and Strauss.

Now consider the hypothesis of our main result, that there is a P- or QP-martingale that succeeds on P/poly $\cap$ EXP. Reading from part (a) of Theorem 17, we would conclude that there is a P/poly- or QP/poly-natural property Π that diagonalizes against P/poly $\cap$ EXP. But our key Lemma 9, combined with Lemma 10, actually yields a much stronger conclusion, namely that the constructed Π diagonalizes i.o. against all of P/poly. This is what is needed for the Razborov-Rudich result.

It is precisely this kind of strengthening that we have been unable to obtain, on hypothesis that there is a $\Gamma_2(\mathrm{P})$ martingale that succeeds on $\mathrm{AC}^0[2] \cap \mathrm{P}$, although Appendix 2 presents some ideas that make this plausible. What we have is that no $\Gamma_2(\mathrm{P})$ martingale can succeed on all of (nonuniform) $\mathrm{AC}^0[2]$.

This point is important because there *does* exist a P/poly-natural proof against P/poly $\cap$ EXP, indeed against any given recursively presentable class $\mathcal{C}$. Let $Q_1, Q_2, \ldots$ be a recursive enumeration of $\mathcal{C}$-machines. Given n, define

$$\Upsilon_n = \{w \in F_n : (\exists i \leq n) w = L(Q_i)^{=n}\},$$

and put $\Pi_n := F_n \setminus \Upsilon_n$. Then $\Pi \in$ P/poly because for strings of length n, i.e. for w of length $N = 2^n$, we can "hard-wire" the n-many characteristic sequences of how machines $Q_1, \ldots, Q_n$ behave at length n. Also each Π_n has density $1 - n/2^N$, which is huge. And Π diagonalizes against $\mathcal{C}$, in fact diagonalizing a.e.

This shows that having a P/poly-natural proof that diagonalizes against P/poly $\cap$ EXP does not suffice for the Razborov-Rudich result. It follows from results in the next section that this Π can be converted into a P/poly-martingale that succeeds on P/poly$\cap$EXP. Hence it is important for our main theorem that the martingale succeeding on P/poly $\cap$ EXP is computable in a uniform complexity class.

5. Are martingales and natural properties equivalent?

The underlying idea behind the concepts of martingales and natural properties is a strong form of diagonalization, and it is natural to ask whether they are equivalent. In this section, we prove a partial converse of Theorem 17. Our results emphasize that two parameters in the definition of natural properties that are somewhat submerged in [26, 27] are very important: the *density* $\rho(\Pi_n)$ of the property Π, that is, $\|\Pi_n\|/\|F_n\|$, and whether Π diagonalizes i.o. or a.e. (see Equations 3 and 4).

Our results here seem to need somewhat stronger closure properties of the class $\mathcal{D}$. Say a circuit class $\mathcal{D}$ is *nice* if it is closed under parallel evaluation of polynomially many functions in $\mathcal{D}$, under finite composition, and under the operation of finding "majority." Clearly P/poly is a nice circuit class.

Theorem 18. *Let $\mathcal{D}$ be a nice class, and let $\mathcal{C}$ be any class of languages. Then:*

(a) If there is a natural property $\Pi \in \mathcal{D}$ of density $1/n = 1/(\log N)$ that diagonalizes a.e. against $\mathcal{C}$, then there is a martingale computable in $\mathcal{D}$ that succeeds on $\mathcal{C}$.

(b) If there is a natural property $\Pi \in \mathcal{D}$ of density $(1 - 1/n^{1+\varepsilon}) = (1 - 1/(\log N)^{1+\varepsilon})$ that diagonalizes i.o. against $\mathcal{C}$, then there is a $\mathcal{D}$-martingale that succeeds on $\mathcal{C}$.

In the case where $\mathcal{D}$ is a uniform complexity class, what the proof gives us is a martingale computable in "randomized $\mathcal{D}$" with bounded (i.e., vanishing) error probability.

Proof. Suppose we have a $\mathcal{D}$-natural property Π that diagonalizes a.e. over $\mathcal{C}$, and let $A = \{A_n\}$ denote the algorithm (family of circuits) that decides Π. For every n, consider the full binary tree T_n of depth $N = 2^n$ that has 2^N leaves in one-to-one correspondence with the members of F_n. Let $\Upsilon_n = F_n \setminus \Pi_n$, and when n is fixed or understood, let $\sigma = \|\Upsilon_n\|/2^N$ denote the density of Υ_n.

For each n, the property $\Pi_n \subseteq F_n$ identifies a large subset of the leaves that are "avoided" by languages in $\mathcal{C}$. By the a.e. diagonalization condition, this means that for every $L \in \mathcal{C}$, and all but finitely many n, L goes through a branch in Υ_n at length n. This is the only property of $\mathcal{C}$ that is used in the proof; the martingale works only with the information about Π_n versus Υ_n. Given unit capital at the root of T_n, the martingale we construct will adopt the following simple strategy: try to make profit along the paths to all leaves in Υ_n, avoiding the leaves in Π_n. By the restriction on information, we allow that there may be no way for the martingale to distinguish among the leaves in Υ_n, so the best it can achieve is to amass a capital of $2^N/\|\Upsilon_n\| = 1/\sigma$ at every leaf in Υ_n.

Suppose the martingale is at an interior node v of T_n. Let $V_0 = \{w \in F_n \mid w \sqsupseteq v0\}$ and $V_1 = \{w \in F_n \mid w \sqsupseteq v1\}$ denote the set of leaves in the subtrees $v0$ and $v1$, respectively. Let $p_0(v) = \|V_0 \cap \Upsilon\|/\|V_0\|$, $p_1(v) = \|V_1 \cap \Upsilon\|/\|V_1\|$. If the martingale could calculate $p_0(v)$ and $p_1(v)$ *exactly*, then it could set $d(v0) = 2d(v)\left(\frac{p_0}{p_0+p_1}\right)$ and $d(v1) = 2d(v)\left(\frac{p_1}{p_0+p_1}\right)$. This would ensure that each leaf in Υ ends up with a capital of $1/\sigma$ (as per the "density systems" idea of Lutz [15]).

The problem is that a martingale that runs in time poly(N) cannot compute the membership in Υ_n of all the 2^N leaves. However, by taking polynomially many random samples at each interior node, a *randomized* machine M can (with high probability) *estimate* the values $p_0(v)$ and $p_1(v)$ to a high degree of accuracy. Then M can use these estimates in lieu of the actual values, and still obey the condition (1) that defines a martingale. This strategy is continued so long as the subtree below v has more than N^2 nodes; when the subtree has atmost N^2 nodes, an exhaustive examination of all leaves is done and most of the capital is diverted towards the leaves in Υ_n, leaving a tiny portion for the leaves in Π_n. This tiny amount is donated to ensure that leaves $z \in \Pi_n$ do not go to zero, so that the martingale may eventually succeed on languages $L \in \mathcal{C}$ with $z \sqsubseteq \chi_L$. To simplify the description of

M and the calculations below, we assume that if M discovers that small subtree with N^2 nodes has *no* leaves that belongs to Υ_n, it chooses some leaf arbitrarily and directs profits toward it. This "wastage" does not matter much to the profits on leaves that actually do belong to Υ_n.

Let $q_0(v)$ and $q_1(v)$ denote, respectively, the estimates of $p_0(v)$ and $p_1(v)$ that are obtained by sampling. Via standard Chernoff-bound methods, one can show that upon taking $\mathrm{poly}(N)$-many samples (for a suitably large polynomial), with probability $1-\exp(-N)$, the estimates are within an additive term of $\delta = 1/\mathrm{poly}(N)$ of the true values. The martingale will then adopt the policy that overestimation (by upto δ) is harmless, but underestimation is dangerous. More precisely, the martingale will pretend that $q_0(v)$ and $q_1(v)$ underestimate $p_0(v)$ and $p_1(v)$, and will therefore use $q_0(v)+\delta$ and $q_1(v)+\delta$ as safer approximations to the actual values. It follows that

$$\frac{d(v0)}{d(v)} = 2\frac{q_0(v)+\delta}{(q_0(v)+\delta)+(q_1(v)+\delta)}, \qquad \frac{d(v1)}{d(v)} = 2\frac{q_1(v)+\delta}{(q_0(v)+\delta)+(q_1(v)+\delta)},$$

and that $d(v0)+d(v1) = 2d(v)$.

Let $m = \lceil 2^N/N^2 \rceil$, let $\tau_1, \tau_2, \ldots, \tau_m$ denote the subtrees of T_n at height $2\log N$ that contain N^2 leaves each. For each i, let u_i denote the root of τ_i, and let p_i denote the probability $\|\mathtt{leaves}(\tau_i)\cap\Upsilon\|/N^2$. Let ρ_i denote the density $\|\mathtt{leaves}(\tau_i)\cap\Upsilon\|/\|\Upsilon\|$; it is easy to see that $\rho_i = \frac{p_i}{p_1+p_2+\ldots+p_m}$. The total value of $d(\cdot)$ at height $2\log N$ is exactly $2^{N-2\log N} = m$, and the strategy works if for each i, $d(u_i) = \Omega(\rho_i m)$. We show:

Claim. For every i, $d(u_i) \geq 0.99\rho_i m$ whp.

Wlog. let $i=1$, and focus on the first subtree τ_1 with N^2 leaves. Recall that by the simplifying assumption made above, for all i, $p_i \geq 1/N^2$. The worst case for τ_1 is the following: at every ancestor v of τ_1, the subtree of v containing τ_1 had an underestimated probability, and the other subtree of v had an overestimated probability. To wit: at the first level, p_1 is underestimated to be $p_1-\delta$, and p_2 is overestimated to be $p_2+\delta$; at the second level, $\frac{1}{2}(p_1+p_2)$ is underestimated to be $\frac{1}{2}(p_1+p_2)-\delta$, and $\frac{1}{2}(p_3+p_4)$ is overestimated to be $\frac{1}{2}(p_3+p_4)+\delta$, and so on. When this happens,

$$\begin{aligned}
d(u_1) &\geq \left(\frac{p_1-\delta+\delta}{(p_1-\delta+\delta)+(p_2+\delta+\delta)}\right)2d(\mathtt{parent}(u_1)) \\
&= 2\left(\frac{p_1}{p_1+p_2+2\delta}\right)d(\mathtt{parent}(u_1)) \\
&\geq 4\left(\frac{p_1}{p_1+p_2+2\delta}\right)\left(\frac{p_1+p_2}{p_1+p_2+p_3+p_4+4\delta}\right)d(\mathtt{parent}(\mathtt{parent}(u_1))) \\
&\ \vdots \\
&\geq m\left(\frac{p_1}{p_1+p_2+2\delta}\right)\left(\frac{p_1+p_2}{p_1+p_2+p_3+p_4+4\delta}\right)\cdots\left(\frac{p_1+\ldots+p_{m/2}}{p_1+\ldots+p_m+m\delta}\right).
\end{aligned}$$

Multiplying and dividing the above by $(p_1+\ldots+p_m)$, and regrouping the terms,

$$d(u_1) \geq m\left(\frac{p_1}{p_1+\ldots+p_m}\right)\left(\frac{p_1+p_2}{p_1+p_2+2\delta}\right)\cdots\left(\frac{p_1+\ldots+p_m}{p_1+\ldots+p_m+m\delta}\right)$$

$$
\begin{aligned}
&= m\rho_1 \left(1 - \frac{2\delta}{p_1 + p_2 + 2\delta}\right) \cdots \left(1 - \frac{m\delta}{p_1 + \ldots + p_m + m\delta}\right) \\
&\geq m\rho_1 \prod_{\ell=1}^{\log m} \left(1 - \frac{2^\ell \delta}{2^\ell p + 2^\ell \delta}\right) && \text{recalling that for all } i,\ p_i \geq 1/N^2 = p \\
&= m\rho_1 \left(1 - \frac{\delta}{p + \delta}\right)^{\log m} \\
&\geq m\rho_1 \left(1 - \frac{1}{N^2}\right)^{N} && \text{setting } \delta = 1/N^4 \\
&= m\rho_1 e^{-1/N} \\
&\geq 0.99 m\rho_1 && \text{for } N \geq 100.
\end{aligned}
$$

♣

By standard arguments about converting high-probability algorithms into non-uniform algorithms, this can be shown to give a $\mathcal{D}/$ poly martingale that succeeds on $\mathcal{C}$.

If the only information used by the martingale is the fact that for every L in $\mathcal{C}$, $L^{=n} \in \{0,1\}^N \setminus \Pi_n$ (io/ae), then the factor of $1/\sigma = 1/(1 - \rho(\Pi_n))$ is the best possible in stage n. If Π is a.e. diagonalizing, then a density of $\Omega(1/n) = \Omega(1/\log N)$ for Π_n gives a factor of $\Omega(1 + 1/n)$ in stage n, which suffices for the martingale to succeed on $\mathcal{C}$.

If Π is merely i.o. diagonalizing, then the above factor seems insufficient. By a modification of the Borel-Cantelli lemma as applied to martingales [15] (see also [28]), it can be shown that if $\sum_n (1 - \rho(\Pi_n))$ converges, then a successful martingale of equivalent nonuniform complexity can be constructed. For example, an i.o.-natural property Π of density $1 - \frac{1}{n^{1+\varepsilon}}$ for some $\varepsilon > 0$ against $\mathcal{C}$ would give a non-uniform martingale that succeeds on $\mathcal{C}$. □

5.1. Concluding Remarks

One of the original motivations for this research was to find a sufficient condition for Lutz's hypothesis $\neg\mu(\mathrm{NP}|\mathrm{EXP}) = 0$. We briefly analyze whether Theorem 11 can be made to work with NP in place of P/poly. Our proof works by taking a hard PSRG G and a *given* QP-computable martingale d, and constructing a language $L \in \mathrm{P/poly} \cap \mathrm{EXP}$ on which d does not succeed. The languages L involved are defined by non-uniform sequences of seeds x for the "amplified generator" $f_x = G_x(y)$ defined from G in [26, 7]. These seeds define the circuits C_n in our key Lemma 9. The selection of sequences C_n in Lemma 9 is non-uniform, however.

We remark that Rudich [personal communication] has recently given evidence based on other beliefs about PSRGs that there cannot be an *NP*-natural property against P/poly. To use his new result to derive a contradiction from $\mu(\mathrm{NP}|\mathrm{EXP}) = 0$, one needs to find a way to transfer nondeterminism from the class on which the martingale succeeds to the martingale itself, trading non-determinism for non-uniformity in the class covered. We leave this as an interesting problem, and also leave the problem of whether the existence of secure PSRGs implies that NP does not have measure zero in E.

We have shown that there is much ground for a deeper investigation into details of the natural-proofs theory of [26], in terms of the *size* of the properties and whether the diagonalization is i.o. or a.e. This may have further ramifications for the connections to formal systems shown by Razborov [25]. Finally, the idea of "randomized martingales" used to prove Theorem 18, and that of "honest" martingales that bypass the non-uniformity problem, seem to merit further study in themselves.

Acknowledgments We are especially grateful to Alexander Razborov for sending us preprints of his recent relevant work, including the journal version of the "Natural Proofs" paper, and to Stephen Rudich for personal communications about extensions of their work. Razborov, Rudich, Eric Allender, Jack Lutz, Elvira Mayrodomo, Alan Selman, and Martin Strauss have given us helpful comments on earlier drafts of this paper.

References

[1] E. Allender and M. Strauss. Measure on small complexity classes, with applications for BPP. Technical Report DIMACS TR 94-18, Rutgers University and DIMACS, April 1994.

[2] E. Allender and M. Strauss. Measure on small complexity classes, with applications for BPP. In *Proc. 35th FOCS*, 1994. to appear.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and hardness of approximation problems. In *Proc. 33rd FOCS*, pages 14–23, 1992.

[4] L. Babai, L. Fortnow, L. Levin, and M. Szegedy. Checking computations in polylogarithmic time. In *Proc. 23rd STOC*, pages 21–31, 1991.

[5] R. Boppana and R. Hirschfeld. Pseudorandom generators and complexity classes. In S. Micali, editor, *Randomness and Computation*, volume 5 of *Advances in Computing Research*, pages 1–26. JAI Press, Greenwich, CT, USA, 1989.

[6] J. Doob. *Measure Theory.* Springer Verlag, New York, 1991.

[7] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *J. ACM*, 33:792–807, 1986.

[8] J. Hastad. Pseudorandom generation under uniform assumptions. In *Proc. 22nd STOC*, pages 395–404, 1990.

[9] J. Hastad, R. Impagliazzo, L. Levin, and M. Luby. Construction of a pseudo-random generator from any one-way function. Technical Report 91–68, International Computer Science Institute, Berkeley, 1991.

[10] R. Impagliazzo, L. Levin, and M. Luby. Pseudo-random generation from one-way functions (extended abstract. In *Proc. 21st STOC*, pages 12–24, 1989.

[11] R. Impagliazzo and M. Naor. Efficient cryptographic schemes provably as secure as subset-sum. In *Proc. 30th FOCS*, 1989.

[12] D. Juedes and J. Lutz. The complexity and distribution of hard problems. In *Proc. 34th FOCS*, pages 177–185, 1993. SIAM J. Comput., to appear.

[13] S. Kautz and P. Miltersen. Relative to a random oracle, NP is not small. In *Proc. 9th Structures*, pages 162–174, 1994.

[14] J. Lutz. A pseudorandom oracle characterization of BPP. In *Proc. 6th Structures*, pages 190–195, 1991.

[15] J. Lutz. Almost everywhere high nonuniform complexity. *J. Comp. Sys. Sci.*, 44:220–258, 1992.

[16] J. Lutz. The quantitative structure of exponential time. In *Proc. 8th Structures*, pages 158–175, 1993.

[17] J. Lutz and E. Mayordomo. Measure, stochasticity, and the density of hard languages. In *Proc. 10th STACS*, volume 665 of *Lect. Notes in Comp. Sci.*, pages 38–47. Springer Verlag, 1993.

[18] J. Lutz and E. Mayordomo. Cook versus Karp-Levin: Separating completeness notions if NP is not small. In *Proc. 11th STACS*, volume 775 of *Lect. Notes in Comp. Sci.*, pages 415–426. Springer Verlag, 1994.

[19] J. Lutz and E. Mayordomo. Measure, stochasticity, and the density of hard languages. *SIAM J. Comput.*, 23:762–779, 1994.

[20] E. Mayordomo. Almost every set in exponential time is P-bi-immune. In *Proc. 7th MFCS*, volume nnn of *Lect. Notes in Comp. Sci.*, pages 392–400. Springer Verlag, 1992. Theor. Comp. Sci., to appear.

[21] E. Mayordomo. *Contributions to the Study of Resource-Bounded Measure.* PhD thesis, Universidad Polytécnica de Catalunya, Barcelona, April 1994.

[22] N. Nisan. Pseudorandom bits for constant-depth circuits. *Combinatorica*, 11:63–70, 1991.

[23] J. Oxtoby. *Measure and Category.* Springer Verlag, New York, 2nd edition, 1980.

[24] K.R. Parthasarathy. *Introduction to Probability and Measure.* The Macmillan Company of India, Ltd., Madras, 1977.

[25] A. Razborov. Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic. *Izvestiya of the RAN*, 1994. To appear.

[26] A. Razborov and S. Rudich. Natural proofs. In *Proc. 26th STOC*, pages 204–213, 1994.

[27] A. Razborov and S. Rudich. Natural proofs, 1994. Update of STOC paper, November 1994.

[28] K. Regan and D. Sivakumar. Improved resource-bounded Borel-Cantelli and stochasticity theorems. Technical Report UB-CS-TR 95-07, Computer Science Dept., University at Buffalo, February 1995.

[29] C.P. Schnorr. A unified approach to the definition of random sequences. *Math. Sys. Thy.*, 5:246–258, 1971.

[30] C.P. Schnorr. *Zufälligkeit und Wahrscheinlichkeit*, volume 218 of *Lect. Notes in Math.* Springer Verlag, 1971.

[31] C.P. Schnorr. Process complexity and effective random tests. *J. Comp. Sys. Sci.*, 7:376–388, 1973.

[32] M. Sudan. *Efficient checking of polynomials and proofs and the hardness of approximation problems*. PhD thesis, University of California, Berkeley, 1992.

Appendix 1

Proof sketch of Theorem 7.

(This is only to bridge the gap between the result stated in [25] and the proof of the weaker result given in [26].) For part (a), we first note the following, which is implicit in [26].

Lemma 19. *If a natural property* Π *(of arbitrary complexity) diagonalizes over* P/poly*, then for every polynomial* q*, there exist infinitely many* n *such that for every circuit* C_n *of size at most* $q(n)$*,* $L(C_n)^{=n}$*, treated as a* 2^n*-bit string, does not belong to* Π_n*.*

Proof. Suppose to the contrary that for some polynomial q there exists $n_0 \geq 0$ such that for all $n \geq n_0$, there exists a circuit C_n of size $q(n)$ such that $L(C_n)^{=n} \in \Pi_n$. Define a language L by letting $L^{=n} = L(C_n^*)^{=n}$ for all $n \geq n_0$, where C_n^* denotes the lexicographically first circuit of size $p(n)$ that satisfies $L(C_n^*)^{=n} \in \Pi_n$. Clearly $L \in$ P/poly, yet Π does not diagonalize over L, a contradiction. □

Now for Theorem 7(a), let a PSRG G and an arbitrary $\varepsilon > 0$ be given. The goal is to show that for infinitely many k, $H(G_k) \leq 2^{k^\varepsilon}$. Let the natural property Π against P/poly be such that each Π_n has density $1/2^{(\log N)^c}$ and circuit size $2^{(\log N)^c} = 2^{n^c}$. For any n, set $k = n^{c/\varepsilon}$. Using G, one can build a *pseudorandom function generator* [7] f as follows: given a seed x of size k, a (pseudorandom) Boolean function $f_x : \{0,1\}^n \to \{0,1\}$ is defined such that there is a circuit of size $\mathrm{poly}(n^{c/\varepsilon}) = \mathrm{poly}(n)$ that computes $f_x(y)$ for all $y \in \{0,1\}^n$. Using this construction, every infinite sequence of seeds $\vec{x} = x_1, x_2, \ldots$ gives a language $L_{\vec{x}}$, and all such languages have circuit families of a fixed polynomial size, say $q(n)$.

Now by Lemma 19, there are infinitely many n such that for every seed x, $f_x \notin \Pi_n$. On the other hand, by the largeness of Π, it follows that a randomly chosen $f \in \{0,1\}^{2^n}$ belongs to Π_n with probability at least $1/2^{O(n^c)}$. This shows that a circuit for Π_n is a statistical test of size $2^{O(n^c)} = 2^{O(k^\varepsilon)}$ that distinguishes f_x from a truly random Boolean function f. The remaining details are the same as in [26] drawing on [7]: Using this statistical test, one can build a statistical test of the same size that distinguishes (with bias of the same order) the output of G_k from a truly random string of length $2k$. Since ε was chosen to be arbitrary, the result follows. For the sake of completeness, we show how this conversion is done.

Claim. Suppose there is a circuit C_n of size $2^{O(n^c)}$ that achieves a bias of $2^{-O(n^c)}$ in distinguishing between f_x when x is chosen randomly from $\{0,1\}^k$, $k = n^{c/\varepsilon}$ and a randomly chosen 2^n-bit string. Then there is a circuit D_k of size $2^{O(n^c)} = 2^{k^\varepsilon}$ that achieves a bias of $2^{-O(n^c)} = 2^{-k^\varepsilon}$ in distinguishing between $G(x)$ when x is chosen randomly from $\{0,1\}^k$, and a randomly chosen $2k$-bit string.

Proof of Claim. Consider the full binary tree T of height n. Label the internal nodes of T by $v_1, v_2, \ldots, v_{2^n-1}$ such that if v_i is a child of v_j then $i < j$. Note that T has 2^n leaves; we will associate the leaves in one-to-one correspondence with all strings of length n. Denote by T_i the union of subtrees of T consisting of the nodes $v_1, \ldots, v_i$, together with all leaves. For a leaf y of T let $v_i(y)$ be the root of the subtree in T_i containing y. For all leaves y, define $G_{0,y}$ to be the identity function, and let $G_{i,y}$ denote the composition $G_{y_n} \circ G_{y_{n-1}} \cdots G_{y_{n-h(i,y)+1}}$. Here $h(i,y)$ denotes the height of y in T_i, or the distance between $v_i(y)$ and y. To each internal node v of the tree T, assign a string x_v chosen uniformly at random from $\{0,1\}^k$. Next, define the random collection f_i to be the collection of functions $\{f_{i,x}\}$ described as follows. Let z be a leaf of the tree. Define $f_{i,x}(z)$ to be the first bit of $G_{i,z}(x_{v_i(z)})$. Note that f_0 is just a random boolean function on n variables, and f_{2^n-1} is just f_x defined above. We know that

$$|\Pr[C_n(f_0) = 1] - \Pr[C_n(f_x) = 1]| \geq 2^{-O(n^c)}.$$

Therefore, there must exist an index i such that

$$|\Pr[C_n(f_i) = 1] - \Pr[C_n(f_{i+1}) = 1]| \geq 2^{-O(n^c)}.$$

At this point, an averaging argument shows that we can fix all the random strings assigned to the nodes of T except the children of v_{i+1} while preserving the bias. (This might determine many of the bits of f_x.) Now there are two ways of assigning strings to the children of v_{i+1}: either assign them both independently chosen random strings from $\{0,1\}^k$, or assign a random string u to v_{i+1} and assign to its two children the strings $G_0(u)$ and $G_1(u)$ respectively. The crucial observation we make is that if these two nodes are assigned strings in the first way, then the resulting boolean function induced on the leaves is precisely f_i, and if they are assigned strings in the second way, then the resulting boolean function induced on the leaves is precisely f_{i+1}. To complete the proof, we will build a circuit D_n that takes a string in $\{0,1\}^{2k}$ and computes the resulting boolean function at the leaves (which one of f_i or f_{i+1}) as described, and feeds the result (f_i or f_{i+1}) to C_n. Note that computing f_i or f_{i+1} can be done in time $2^n \cdot \mathrm{poly}(n)$. Therefore, the size of D_n is bounded by $2^{O(n^c)}$. Now, C_n has an advantage of at least $2^{-O(n^c)}$ in distinguishing between f_i and f_{i+1}, whence it follows that $H(G_k)$ is bounded by $2^{O(n^c)} = 2^{O(k^\epsilon)}$. □

The adjustments required to prove part (b) use the fact that for any h, using Nisan's construction [22], one can build a pseudorandom function generator computable in $\mathrm{AC}^0[2]$ that is secure against depth h circuits of quasipolynomial size. □

Appendix 2: On the measure of $\mathrm{AC}^0[2]$

(This is taken from a draft of October 1994, and will be shortened in journal copy.)

Allender and Strauss [1, 2] define two notions of measure on P. Let $\mu_{\Gamma(\mathrm{P})}$ denote either notion of measure defined in their paper. Using the ideas in the proof of our main theorem together with some improvements, we will show that if $\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2]) = 0$, then pseudo-random generators of certain strength secure against depth 3 q-AC^0 circuits do not exist. Here q-AC^0 denotes the class of languages recognized by a family of constant depth, quasi-polynomial size circuits. However, Nisan [22] has shown that such a generator does exist, and it follows that $\neg(\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2]) = 0)$. As in case of P/poly, it is easy to see that one can construct a q-AC^0-natural property against $\mathrm{AC}^0[2]$ from a $\Gamma(\mathrm{P})$-martingale that succeeds on $\mathrm{AC}^0[2]$.

Remark: The definitions for the $\Gamma(\mathrm{P})$ measures and the details of Nisan's generator are fairly technical. Moreover, the parameters we use for Nisan's generator are not the same as those used by [26]. For this reason, we will not prove our theorem by constructing the natural property and appealing to [26]. Instead, we present a complete proof of the result "$\neg(\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2]) = 0)$" by supplying all the arguments in careful detail.

Before analyzing the notions of "measure within P" defined by Allender and Strauss, we find it convenient to change the way machines computing martingales $d(w)$ are described. The new formalism is essentially the same as that for "holographic proofs" in [4, 3, 32] with w playing the role of the "proof." Namely, define a *query machine* M to have a standard TM input tape, any number of standard worktapes, and a *query tape* that provides "random-access" to bits of a string w given as an auxiliary input. M is given as input the length N of w in standard dyadic notation, and is allowed to write integers $i \leq N$ on its query tape, receiving in answer the bit w_i. The string N is the same as the string x_N whose membership or non-membership in languages with initial segment w is indexed by the last bit of w. The main change is that now complexity bounds are expressed in terms of $n = |x_N| = \lceil \lg N \rceil$ rather than N. Thus Lutz's martingales for EXP are exactly those computable in time 2^{n^c} for some fixed $c > 0$, and the P-martingales for measure in E are those in time $2^{O(n)}$.

Here we are interested in time polynomial in n, or equivalently, polylogarithmic in N. Allender and Strauss [2] note that this alone may not yield a non-trivial measure on P, and give reasons for adding the following restriction on "dependency set size." In our scaling, this becomes:

Definition 6 (cf. [2]). Let M be a query machine that computes a function f. For all inputs N and auxiliary inputs w of length N, define

$$Y(w) = \{i : 1 \leq i \leq N, M \text{ queries bit } w_i \text{ in the computation of } f(w)\}.$$

A set $S_N \subseteq \{1, \ldots, N\}$ is a *dependency set for length* N if for all $K \in S_N$ and *all* w of length K, $Y(w) \subseteq (S_N \cap \{1, \ldots, K\})$. Finally, M is said to have *polylogarithmic dependency set size* if there exists a polynomial p such that for all N, M has a dependency set S_N for length N of size at most $p(\log N)$.

As noted by Allender and Strauss, there is always a minimum dependency set S_N for each N. Recalling that $N = 2^n$, the point is that the size of S_N is polynomial in n. We also say that the functon f computed by M has polylogarithmic dependency set size.

Allender and Strauss [2] offer two notions of measure in terms of $\Gamma(\mathrm{P})$-machines, and we need to examine the following technicalities of resource-bounded measure theory to appreciate them. The following should be contrasted with the conditions (1) and (2) in Section 2.2.

Definition 7. A *super-martingale* is a function d from $\{0,1\}^*$ into the nonnegative reals that satisfies the following "inexact average law": for all $w \in \{0,1\}^*$,

$$d(w) \geq \frac{d(w0) + d(w1)}{2}. \tag{6}$$

Regarded as a betting strategy, a super-martingale is allowed to "throw away money" when $d(w) > (d(w0) + d(w1))/2$. The *success class* $\underline{S}^*[d]$ of a super-martingale d is defined to be the class of languages A such that $\limsup_{w \sqsubseteq A} d(w) = +\infty$, or equivalently,

$$(\forall K > 0)(\exists w)[w \sqsubseteq A \ \wedge \ d(w) \geq K]. \tag{7}$$

Definition 8 ([2]). Allender and Strauss define the following notions of measure in P:

(a) Write $\mu_{\Gamma(\mathrm{P})}(\mathcal{C}) = 0$, and call $\mathcal{C}$ $\Gamma(\mathrm{P})$-*null* if there is a $\Gamma(\mathrm{P})$-machine that computes a martingale $d : \{0,1\}^* \to \mathbf{D}$ such that $\mathcal{C} \subseteq S^*[d]$.

(b) Write $\underline{\mu}_{\Gamma(\mathrm{P})}(\mathcal{C}) = 0$ if there is a $\Gamma(\mathrm{P})$-machine that computes a super-martingale $d : \{0,1\}^* \to \mathbf{D}$ such that $\mathcal{C} \subseteq \underline{S}^*[d]$.

Also write $\mu(\mathcal{C}|\mathrm{P}) = 0$ if $\mu_{\Gamma(\mathrm{P})}(\mathcal{C} \cap \mathrm{P}) = 0$, and $\mu(\mathcal{C}|\mathrm{P}) = 1$ if $\mu_{\Gamma(\mathrm{P})}((\mathrm{P} \setminus \mathcal{C})|\mathrm{P}) = 0$.

Allender and Strauss have shown (see [2]) that relaxing d in (a) to be a super-martingale, or leaving d a martingale and allowing $\mathcal{C} \subseteq \underline{S}^*[d]$, does not change (a). They also show robustness under adopting Lutz's original terms of *density systems* $d : \mathbf{N} \times \{0,1\}^* \to \mathbf{R}$ and *approximate* computations of d. The one lack of robustness between the measures $\mu_{\Gamma(\mathrm{P})}$ and $\underline{\mu}_{\Gamma(\mathrm{P})}$ is shown graphically by the following proposition. A language L is said to be P-*printable* if there is a Turing Machine M that, on input 0^n outputs the list of all strings in $L \cap \{0,1\}^n$. Note that P-printable languages are both polynomially *sparse* and have P-uniform AC^0 circuits of depth two. (The following proposition is due to Allender and Strauss [unpublished, personal communications 8/94—11/94]; parts (b) and (c) were obtained independently by us.)

Proposition 20.

(a) *There exists a* $\Gamma(\mathrm{P})$*-computable super-martingale* d *such that every language of density* $< 2^n/n^3$ *belongs to* $S^*[d]$*. In particular,* $\underline{\mu}(\mathrm{SPARSE}|\mathrm{P}) = 0$.

(b) *For every* $\Gamma(\mathrm{P})$*-computable super-martingale* d*, there is a* P*-printable language* A *such that* $A \notin S^{\infty}[d]$*; i.e., on which* d *does not succeed in the sense of a limit.*

(c) *For every* $\Gamma(\mathrm{P})$*-computable martingale* d*, there is a* P*-printable language* A *such that* $A \notin S^{\infty}[d]$*, that is,* d *does not succeed on* A *in the sense of a limit.*

The proof idea of (a) is to divide each segment $\{0,1\}^n$ into intervals of size n^3, and assign each interval I a "base value" $e(I)$ that decreases very slowly. The $\Gamma(\mathrm{P})$-machine M only queries bits in the interval containing its input N, so its dependency sets have size n^3. On all branches except the all-0 branch it outputs $e(I)$, while on the all-0 branch it outputs $e(I-1) + [e(I-1) - e(I)] \cdot 2^{r-1}$, where r is the distance from N to the left boundary of its interval. Every language A of small density must fall into an all-0 branch of infinitely many intervals, and careful choice of $e(\cdot)$ makes the values in these intervals unbounded. Strict inequality in (6) holds at interval boundaries, and success is by lim sup.

Intuitively, the difference between the measure $\mu_{\Gamma(\mathrm{P})}$ and the more-relaxed measure $\underline{\mu}_{\Gamma(\mathrm{P})}$ is that, given a super-martingale d, attempting to enforce either the exact average law or the limit success condition $\mathcal{C} \subseteq S^*[d]$ can blow up the dependency set size by an amount exponential in n.

Allender and Strauss [2] show that P-uniform AC^0 is not measurable by $\Gamma(\mathrm{P})$ martingales with the "limit" notion of success. Below, we show that non-uniform $\mathrm{AC}^0[2]$, that is, the class of languages recognized by a family of polynomial-sized, constant depth circuits using $\{\wedge, \vee, \neg, \oplus\}$-gates, does not have $\Gamma(\mathrm{P})$ measure zero, even under the more liberal definition of success by the upper limit. This also implies that non-uniform NC^1 does not have $\Gamma(\mathrm{P})$ measure zero under the upper limit notion of success. Our intent is to combine the technique of Section 3 with the strong pseudorandom generators for constant-depth circuits constructed by Nisan [22].

Theorem 21 ([22]). *Let* $c, h \geq 1$ *be fixed integers; let* $a = 2(ch + c + 1)$*. There is a family of functions* $G = \{G_n : \{0,1\}^{n^a} \to \{0,1\}^{2^n}\}$ *such that for all* n*:*

(1) there is a circuit of size $\mathrm{poly}(n^a)$ *and constant depth with* $\{\wedge, \vee, \neg, \oplus\}$*-gates that, for any seed* $s \in \{0,1\}^{n^a}$ *and any* $y \in \{0,1\}^n$*, determines the bit of* $G_n(s)$ *indexed by* y*.*

(2) letting $N = 2^n$*, for any circuit* C *of size at most* $2^{(\log N)^c} = 2^{n^c}$*, depth* h *with* $\{\wedge, \vee, \neg\}$*-gates,*

$$|\Pr[C(Y) = 1] - \Pr[C(G_n(s)) = 1]| \leq \frac{1}{2^{(\log N)^c}} = \frac{1}{2^{n^c}},$$

where Y *is a string chosen uniformly at random from* $\{0,1\}^{2^n}$*, and* s *is a seed chosen uniformly at random from* $\{0,1\}^{n^a}$*.*

Before we prove the theorem showing that non-uniform $\mathrm{AC}^0[2]$ does not have $\Gamma(\mathrm{P})$ measure zero under the upper limit definition for success, we need slightly improved versions of Lemmas 8 and 9.

Lemma 22. *Let d be a martingale. For any string u and any $\ell \in \mathbf{N}$, $b \in \mathbf{R}$,*

$$\left\| \{v \in \{0,1\}^\ell : (\forall w \sqsubseteq v) d(uw) \leq \left(1 + \frac{1}{b}\right) d(u)\} \right\| \geq 2^\ell \left(\frac{1}{b+1}\right).$$

Proof. It follows from the definition of a martingale that $\sum_{v \in \{0,1\}^\ell} d(uv) = 2^\ell \cdot d(u)$. Suppose by way of contradiction that for some u, ℓ, and b, the inequality in the statement of the lemma does not hold. This implies

$$\left\| \{v \in \{0,1\}^\ell : (\exists w \sqsubseteq v) d(uw) > \left(1 + \frac{1}{b}\right) d(u)\} \right\| \geq 2^\ell \left(\frac{b}{b+1}\right). \tag{8}$$

Consider the complete binary tree T of depth ℓ, and imagine that the root of the tree is endowed with a capital sum of $d(u)$. The interior nodes of T can be associated in one-to-one correspondence with $\{0,1\}^{<\ell}$, and the leaves of T with $\{0,1\}^\ell$ in the obvious way; we will, therefore, refer to the nodes of T directly as strings in $\{0,1\}^{\leq \ell}$. Each node v in the tree (leaves as well as interior nodes) will be annotated by the value $d(uv)$. The annotations describe the *strategy* of martingale d in the obvious way. Call an interior node w of T *rich* if $d(uw) > (1 + 1/b)d(u)$, and call a leaf v of T *prodigal* if v is the descendant of some rich interior node w. Then Equation 8 is the same as saying that the number of prodigal leaves v is greater than $2^\ell \cdot (b/b + 1)$.

For each prodigal leaf v, mark the rich ancestor of v that is closest to the root. Beginning at the root, perform a breadth-first traversal of T, visiting the interior nodes level by level in a top-down fashion. Whenever a marked rich node w is visited during the traversal, annotate the entire subtree of w by the value $d(uw)$, and unmark any marked rich node in this subtree, and call w *frozen*.

Let T' denote the tree when the freeze-as-you-go traversal is complete. It is easy to see that T' represents a valid martingale strategy, since annotating the subtree of node w by $d(uw)$ corresponds to playing a safe strategy (without making any wager) on these strings. More importantly, we claim that every leaf v that was labeled *prodigal* in T is *rich* in T'. To see this, note that if v was prodigal in T, some rich ancestor w of v was marked. Now, there are two cases. If there was no marked rich node on the path from the root to w, then w continues to be rich in T'. If there was at least one marked rich node on the path from the root to w, then let z denote the one closest to the root. By the freezing policy, the value of w in T' equals the value of z on T, which, by definition, is sufficient to keep w rich. Therefore, in either case, w is rich in T'. Again by the freezing policy, v inherits all the wealth of w (no richer, no poorer), so v is rich.

Let d' denote the martingale that behaves exactly like d on all strings of length at most $|u|$, and then adopts the strategy given by T' on extensions of u. Under this strategy, the number of rich leaves in T is at least $2^\ell \cdot (b/b+1)$, and each rich leaf has an annotated value of strictly greater than $(1+1/b)d(u)$. Therefore, the total money that d' has at the bottom of T', namely, $\sum_{v \in \{0,1\}^\ell} d(uv)$, exceeds 2^ℓ, a contradiction. □

Lemma 23. *If a* $\Gamma(\mathrm{P})$ *martingale* d *succeeds on* $\mathrm{AC}^0[2]$, *then for every polynomial* q *and constant* h, *there exist infinitely many* n *and* $\{\wedge, \vee, \neg, \oplus\}$*-circuits* C_i *of size at most* $q(i)$ *and depth at most* h, *for* $0 \le i < n$, *such that for all* $\{\wedge, \vee, \neg, \oplus\}$*-circuits* C_n *of size at most* $q(n)$ *and depth at most* h,

$$(\exists u \sqsubseteq u_n)\left[d(u_0 \ldots u) > \left(1 + \frac{1}{n^2}\right) d(u_0 \ldots u_{n-1})\right],$$

where u_i *is the* 2^i*-bit binary "characteristic string" that indicates the membership in* $L(C_i)$ *of* $\{0,1\}^i$.

Proof. Suppose not. Then there is a polynomial q and constants $h, n_0 \in \mathbf{N}$ such that for all $n \ge n_0$, for every sequence of $\{\wedge, \vee, \neg, \oplus\}$-circuits C_i of size at most $q(i)$, depth at most h, for $0 \le i < n$, there exists a $\{\wedge, \vee, \neg, \oplus\}$-circuit C_n of size at most $q(n)$ and depth at most h such that for every prefix u of u_n, $d(u_1 \ldots u) \le \left(1 + \frac{1}{n^2}\right) d(u_1 \ldots u_{n-1})$, where the u_i's have the same meaning as in the statement of the lemma.

We will build a language L as follows: for strings of length less than n_0, membership in L will be an arbitrary but fixed sequence. Let $\alpha = d(u_1 \ldots u_{n_0-1})$. Clearly $\alpha < \infty$. For $n \ge n_0$, we define $L^{=n}$ inductively. Let $u_1, \ldots, u_{n-1}$ be the result of the recursively applying the construction to obtain $L^{<n}$; that is, $u_i = L^{=i}$. By assumption, there exists a $\{\wedge, \vee, \neg, \oplus\}$-circuit C_n of size at most $q(n)$ and depth at most h, such that for every prefix u of u_n, $d(u_1 \ldots u) \le \left(1 + \frac{1}{n^2}\right) d(u_1 \ldots u_{n-1})$. Set $u_n = L(C^*)^{=n}$, where C^* is the lexicographically first C_n that satisfies this inequality (under some fixed encoding of circuits of the appropriate size, depth and type).

Clearly $L \in \mathrm{AC}^0[2]$, since it can be accepted by the circuit family $[C_n]_{n=0}^{\infty}$. Finally,

$$\limsup_{n \to \infty} d(L^{\le n}) \le \alpha \prod (1 + 1/n^2) < \infty,$$

so d does not succeed on L, a contradiction. □

Theorem 24. $\neg(\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2]) = 0)$.

Proof Sketch. Suppose by way of contradiction that $\mu_{\Gamma}(\mathrm{P})(\mathrm{AC}^0[2]) = 0$. Then there exists a martingale d computable in $\Gamma(\mathrm{P})$ that succeeds on every language in $\mathrm{AC}^0[2]$. Let $f(m) = (\log m)^c$ bound the running time and the dependency set size of a Turing Machine M that computes d.

Similar to the proof of Theorem 11, we can define a family $\{L\}$ of pseudo-random languages in $\mathrm{AC}^0[2]$ as the concatenation of the outputs of the pseudo-random generator on seeds of the appropriate size. Since Nisan's pseudo-random generator can perform an exponential amount of stretching, we can get a characteristic string of length 2^n (corresponding to the membership of all strings of length n) from a seed of size polynomial in n. Let G be the generator from Theorem 21 with parameter $h = 3$ and c from the $\Gamma(\mathrm{P})$-computability of the martingale d. Let $a = 2(ch + c + 1) = 8c + 2$, as described in Theorem 21. For a sequence $s = \{s_n \in \{0,1\}^{n^a}\}$ of seeds, we define $L = L(s)$ by taking $L^{=n}$ to be $G_n(s_n)$. By the computability of G, it follows that L can be accepted by a family of circuits of size $p(n)$ and depth t, comprising $\{\wedge, \vee, \neg, \oplus\}$-gates, for some fixed polynomial p and some fixed constant t. In other words, every such language $L(s)$ is in $\mathrm{AC}^0[2]$.

By Lemma 23 and arguments similar to those of Theorem 11, it follows that we can build a family of circuits for infinitely many n that acts as a statistical test against the generator G. The discussion below is restricted to these "infinitely many n," and the strings u_i have the same meaning as in Lemma 23. Recall that $f(m) = (\log m)^c$ bounds the running time and dependency set size of a Turing Machine that computes the martingale d. On inputs of length 2^n, $f(2^n) = n^c$. For each $Y \in \{0,1\}^{2^n}$ and each $W \sqsubseteq Y$, we can precompute the predicate "$d(u_0 \dots u_{n-1}W) \leq (1+1/n^2)d(u_0 \dots u_{n-1})$." Since the computation of $d(u_0 \dots u_{n-1}W)$ depends on only n^c bits of W, this predicate can be described by a truth table of size $2^{n^c} = 2^{(\log N)^c}$, where, as usual, $N = 2^n$. By hardwiring this truth table as a sum-of-products, we get a circuit C_n of size $2^{O((\log N)^c)}$ and depth 2 using only $\{\wedge, \vee, \neg\}$-gates. Finally, the statistical test "$(\forall W \sqsubseteq Y)[d(u_0 \dots u_{n-1}W) \leq (1 + 1/n^2)d(u_0 \dots u_{n-1})]$" can be computed by taking the AND of each of the above 2^n circuits. Moreover, the statistical test achieves a bias of $1/O(n^2)$ in distinguishing the output of the generator from a truly random string. This contradicts Theorem 21, and it follows that $\neg(\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2]) = 0)$. □

Remarks. While our result is better than that of [2] in one sense (namely, lim sup versus lim), it is inferior in that it deals with $\mathrm{AC}^0[2]$ rather than AC^0 itself, and it lacks the "in P" condition. All our result says is that for any $\Gamma(\mathrm{P})$-computable martingale d, there is a language L in non-uniform $\mathrm{AC}^0[2]$ such that $L \notin S^*[d]$. Equivalently, for any $\Gamma(\mathrm{P})$-computable super-martingale d, there is a language L in non-uniform $\mathrm{AC}^0[2]$ such that $L \notin \underline{S}^*[d]$.

The above proof does not resolve the question of whether $\mu(\mathrm{AC}^0[2]|\mathrm{P}) = 0$ in the negative, since that would entail proving $\mu_{\Gamma(\mathrm{P})}(\mathrm{AC}^0[2] \cap \mathrm{P}) \neq 0$. Compared to Theorem 11, where we obtained the analogous "in EXP" condition, the problem is that while the language L constructed in the proof of Lemma 9 was in EXP-uniform P/poly ($\subseteq$ EXP), the language L constructed here seems not to be in P-uniform $\mathrm{AC}^0[2]$, nor in P. We suspect that it may be possible to exploit the following two properties more fully to obtain this stronger result:

(1) The *polylog-wise independence* of Nisan's generator: Given any set S of indices of the $N = 2^n$ output bits of the generator, such that the size of S is polylog in N (i.e., is polynomial in n), and given any desired setting of the bits indexed by S, there exists a seed string that realizes those bits, and s can be found in time polynomial in n by solving linear equations.

(2) The polynomial-size dependency-set restriction in the μ_2 measure. A closer analysis may yield a stronger version of Lemma 23.