

Random Permutations in Computational Complexity*

John M. Hitchcock[†]Adewale Sekoni[‡]Hadi Shafei[§]

November 9, 2025

Abstract

Classical results of Bennett and Gill (1981) show that with probability 1, $P^A \neq NP^A$ relative to a random oracle A , and with probability 1, $P^\pi \neq NP^\pi \cap coNP^\pi$ relative to a random permutation π . Whether $P^A = NP^A \cap coNP^A$ holds relative to a random oracle A remains open. While the random oracle separation has been extended to specific individually random oracles—such as Martin-Löf random or resource-bounded random oracles—no analogous result is known for individually random permutations.

We introduce a new resource-bounded measure framework for analyzing individually random permutations. We define permutation martingales and permutation betting games that characterize measure-zero sets in the space of permutations, enabling formal definitions of polynomial-time random permutations, polynomial-time betting-game random permutations, and polynomial-space random permutations.

Our main result shows that $P^\pi \neq NP^\pi \cap coNP^\pi$ for every polynomial-time betting-game random permutation π . This is the first separation result relative to individually random permutations, rather than an almost-everywhere separation. We also strengthen a quantum separation of Bennett, Bernstein, Brassard, and Vazirani (1997) by showing that $NP^\pi \cap coNP^\pi \not\subseteq BQP^\pi$ for every polynomial-space random permutation π .

We investigate the relationship between random permutations and random oracles. We prove that random oracles are polynomial-time reducible from random permutations. The converse—whether every random permutation is reducible from a random oracle—remains open. We show that if $NP \cap coNP$ is not a measurable subset of EXP , then $P^A \neq NP^A \cap coNP^A$ holds with probability 1 relative to a random oracle A . Conversely, establishing this random oracle separation with time-bounded measure would imply BPP is a measure 0 subset of EXP .

Our framework builds a foundation for studying permutation-based complexity using resource-bounded measure, in direct analogy to classical work on random oracles. It raises natural questions about the power and limitations of random permutations, their relationship to random oracles, and whether individual randomness can yield new class separations.

*A preliminary version of this paper appeared the *Proceedings of the 50th International Symposium on Mathematical Foundations of Computer Science (MFCS 2025)* [16].

[†]Department of Electrical Engineering and Computer Science, University of Wyoming. jhitchco@uwyo.edu. This research was supported in part by NSF grant 2431657.

[‡]Department of Mathematics, Computer Science & Physics, Roanoke College. sekoni@roanoke.edu

[§]Department of Computer Science, University of Huddersfield. H.Shafei@hud.ac.uk

Contents

1	Introduction	3
1.1	Background	3
1.2	Our Approach: Permutation Martingales and Permutation Measure	3
1.3	Our Results	5
1.4	Random Oracles and Measure 0-1 Laws in EXP	6
1.5	Organization	6
2	Preliminaries	6
3	Permutation Martingales and Permutation Measure	7
3.1	Permutation Measure Space	7
3.2	Permutation Martingales	8
3.3	A Permutation Martingale Example	10
3.4	Permutation Martingales as Random Variables	11
3.5	Resource-Bounded Permutation Measure	11
3.6	Permutation Betting Games	12
3.7	Measure Conservation	12
4	Elementary Properties of Random Permutations	14
4.1	Intractability of Random Permutations	14
4.2	Random Permutations versus Random Oracles	16
5	Random Permutations for $\text{NP} \cap \text{coNP}$	18
6	Random Permutations for $\text{NP} \cap \text{coNP}$ versus Quantum Computation	22
7	Random Oracles for $\text{NP} \cap \text{coNP}$ and 0-1 Laws for Measure in EXP	23
8	Conclusion	26

1 Introduction

The seminal work of Bennett and Gill [5] established two foundational separations in computational complexity theory:

1. $P^A \neq NP^A$ relative to a random oracle A with probability 1.
2. $P^\pi \neq NP^\pi \cap \text{coNP}^\pi$ relative to a random permutation π with probability 1.

Subsequent research extended the first separation to hold for specific, individually random oracles, including algorithmically (Martin-Löf) random oracles [6], polynomial-space-bounded random oracles [22], and polynomial-time betting-game random oracles [15]. However, the second separation has not yet been strengthened in an analogous way. Whether $P^A \neq NP^A \cap \text{coNP}^A$ holds relative to a random oracle A remains an open question.

In this paper, we develop a novel framework for resource-bounded permutation measure and randomness, introducing *permutation martingales* and *permutation betting* games. These concepts generalize classical martingales and betting games to the space Π of all length-preserving permutations $\pi : \{0, 1\}^* \rightarrow \{0, 1\}^*$ where $|\pi(x)| = |x|$ for all $x \in \{0, 1\}^*$.

1.1 Background

Bennett and Gill [5] initiated the study of random oracles in computational complexity, proving that $P^A \neq NP^A$ for a random oracle A with probability 1. Subsequent work extended this to individual random oracles. Book, Lutz, and Wagner [6] showed that $P^A \neq NP^A$ for every oracle A that is algorithmically random in the sense of Martin-Löf [23]. Lutz and Schmidt [22] improved this further to show $P^A \neq NP^A$ for every oracle A that is pspace-random in the sense of resource-bounded measure [20]. Hitchcock, Sekoni, and Shafei [15] extended this result to polynomial-time betting-game random oracles [7].

The complexity class $NP \cap \text{coNP}$ is particularly significant because it comprises problems that have both efficiently verifiable proofs of membership and non-membership. This class includes important problems such as integer factorization and discrete logarithm, which are widely believed to be outside P but are not known to be NP -complete. These problems play a central role in cryptography, as the security of widely-used cryptosystems relies on their presumed intractability [8, 28]. Furthermore, under derandomization hypotheses, $NP \cap \text{coNP}$ has been shown to contain problems such as graph isomorphism [19], further underscoring its importance in complexity theory. Thus, understanding the relationship between P and $NP \cap \text{coNP}$ relative to different notions of randomness could shed light on the structure of these classes and the limits of efficient computation.

1.2 Our Approach: Permutation Martingales and Permutation Measure

In this work, we develop a novel framework for resource-bounded permutation measure and randomness. We introduce permutation martingales and permutation betting games, extending classical notions of random permutations. Our theory captures essential properties of random permutations while enabling complexity separations. We prove that random oracles can be computed in polynomial time from a random permutation; however, the converse remains unresolved.

First, we recall the basics of resource-bounded measure. A martingale in Cantor space may be viewed as betting on the membership of strings in a language. The standard enumeration of $\{0, 1\}^*$ is $s_0 = \lambda, s_1 = 0, s_2 = 1, s_3 = 00, s_4 = 01, \dots$. In the i^{th} stage of the game, the martingale has seen the membership of the first i strings and bets on the membership of s_i in the language. The

martingale's value is updated based on the outcome of the bet. Formally, a classical martingale is a function $d : \{0, 1\}^* \rightarrow [0, \infty)$ satisfying the fairness condition

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

for all strings w . Intuitively, $d(w)$ represents the capital that a gambler has after betting on the sequence of bits in w according to a particular strategy. The fairness condition ensures that the expected capital after the next bit is equal to the current capital. A martingale succeeds on a language $A \subseteq \{0, 1\}^*$ if

$$\limsup_{n \rightarrow \infty} d(A \upharpoonright n) = \infty,$$

where $A \upharpoonright n$ is the length- n prefix of A 's characteristic sequence. The *success set* of d is $S^\infty[d]$, the set of all sequences that d succeeds on. Ville [31] proved that a set X has Lebesgue measure zero if and only if there is a martingale that succeeds on all elements of X . Lutz [20] defined resource-bounded measure by imposing computability and complexity constraints on the martingales in Ville's theorem.

We take a similar approach in developing resource-bounded permutation measure. Unlike a classical martingale betting on the bits of a language's characteristic sequence, a permutation martingale bets on the function values of a permutation π . Instead of seeing the characteristic string of a language, a permutation martingale sees a list of permutation function values. More precisely, after $i \geq 0$ rounds of betting, a permutation martingale has seen a *prefix partial permutation*

$$g = [g(s_0), \dots, g(s_{i-1})]$$

where $|g(s_i)| = |s_i|$ for all i . The permutation martingale will bet on the next function value $g(s_i)$. The current *betting length* is $l(g) = |s_i|$, the length of the next string s_i in the standard enumeration. The set of *free strings* available for the next function value is

$$\text{free}(g) = \left\{ x \in \{0, 1\}^{l(g)} \mid x \text{ is not listed in } g \right\}.$$

For any prefix partial permutation g , a permutation martingale d outputs a value $d(g, x) \geq 0$ for each $x \in \text{free}(g)$. The values satisfy the averaging condition

$$d(g) = \frac{1}{|\text{free}(g)|} \sum_{x \in \text{free}(g)} d(g, x).$$

Here g, x denotes appending the string x as the next function value in prefix partial permutation g . See Figure 1.1 for an example of a permutation martingale betting on strings of length 2.

Prefix partial permutations may be used as cylinders to define a measure in Π that is equivalent to the natural product probability measure. We detail this in Section 3. Briefly, a class $X \subseteq \Pi$ has measure 0 if for every $\epsilon > 0$, there exists a sequence of cylinders $\{\llbracket g_i \rrbracket \mid i \in \mathbb{N}\}$ that has total measure at most ϵ and covers X . This is difficult to work with computationally as the covers may be large and require exponential time to enumerate.

We prove an analogue of Ville's theorem [31], showing that permutation martingales characterize measure 0 sets in the permutation space Π : a class $X \subseteq \Pi$ has measure 0 if and only if there a permutation martingale d with $X \subseteq S^\infty[d]$. This permutation martingale characterization allows us to impose computability and complexity constraints in the same way Lutz did for resource-bounded measure in Cantor space [20]. In the following, let Δ be a resource bound such as $\mathbf{p}$, $\mathbf{p}_2$, $\mathbf{pspace}$, or $\mathbf{p}_2\mathbf{space}$ (see Section 3.5 for more details).

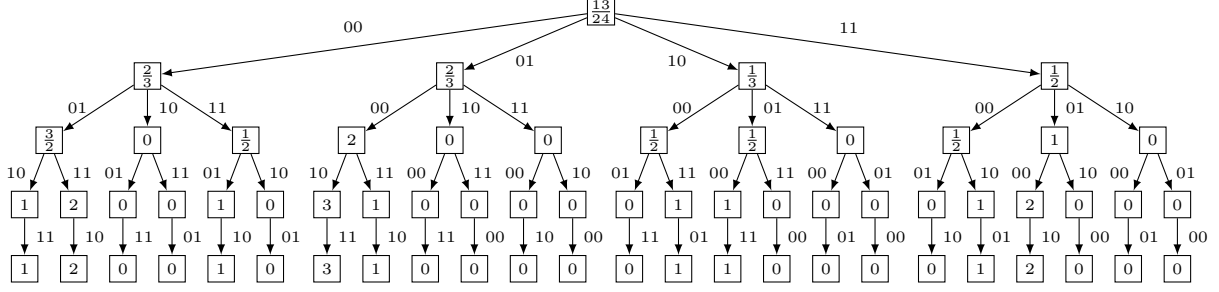


Figure 1.1: An example permutation martingale on strings of length 2. Each path through the tree represents a permutation on $\{00, 01, 10, 11\}$.

Definition 1.1. Let Δ be a resource bound. A class of permutations $X \subseteq \Pi$ has Δ -measure 0 if there is a Δ -computable permutation martingale that succeeds on X .

Betting games [7, 24] are a generalization of martingales that are allowed to bet on strings in an adaptive order rather than the standard order. We analogously introduce permutation betting games as a generalization of both permutation martingales and classical betting games by allowing the betting strategy to adaptively choose the order in which it bets on the permutation's values. We use these betting games to define resource-bounded permutation betting-game measure.

Definition 1.2. Let Δ be a resource bound. A class of permutations $X \subseteq \Pi$ has Δ -betting game measure 0 if there is a Δ -computable permutation betting game that succeeds on X .

We also define *individually* random permutations.

Definition 1.3. Let $\pi \in \Pi$ be a permutation and let Δ be a resource bound.

1. π is Δ -random if no Δ -permutation martingale succeeds on π .
2. π is Δ -betting game random if no Δ -permutation betting game succeeds on π .

1.3 Our Results

Our main result strengthens the Bennett–Gill permutation separation by proving that $P \neq NP \cap \text{coNP}$ relative to every polynomial-time betting-game random permutation π . Formally, Theorem 5.1 establishes that

$$P^\pi \neq NP^\pi \cap \text{coNP}^\pi$$

for every $\mathbf{p}$ -betting-game random permutation π . In fact, we obtain even stronger separations in terms of bi-immunity [3, 9], a notion formalizing the absence of infinite, easily-decidable subsets (see Section 5 for more details). We show that for a $\mathbf{p}$ -betting-game random permutation π , the class $\text{NLIN}^\pi \cap \text{coNLIN}^\pi$ contains languages that are bi-immune to $\text{DTIME}^\pi(2^{kn})$ for all $k \geq 1$, where NLIN denotes nondeterministic linear time. Moreover, relative to a $\mathbf{p}_2$ -betting-game random permutation, we derive that $NP^\pi \cap \text{coNP}^\pi$ contains languages that are bi-immune to $\text{DTIME}^\pi(2^{n^k})$ for every $k \geq 1$.

Bennett et al. [4] showed that $NP^\pi \cap \text{coNP}^\pi \not\subseteq \text{BQTIME}^\pi(o(2^{n/3}))$ relative to a random permutation π with probability 1. We apply our resource-bounded permutation measure framework to improve this to individual space-bounded random oracles. Specifically, we show that relative to a $\mathbf{p}_2\text{space}$ -random permutation π ,

$$NP^\pi \cap \text{coNP}^\pi \not\subseteq \text{BQP}^\pi.$$

This illustrates the power of our framework for analyzing the interplay between randomness, classical complexity, and quantum complexity.

1.4 Random Oracles and Measure 0-1 Laws in EXP

Tardos [29] proved that if $\text{AM} \cap \text{coAM} \neq \text{BPP}$, then $\text{P}^A \neq \text{NP}^A \cap \text{coNP}^A$ with probability 1 for a random oracle A . This is proved using ALMOST complexity classes. For a relativizable complexity class $\mathcal{C}$, its ALMOST- $\mathcal{C}$ class consists of all languages that are in the class with probability 1 relative to a random oracle: $\text{ALMOST-}\mathcal{C} = \{L \mid \Pr[L \in \mathcal{C}^A] = 1\}$. We have $\text{ALMOST-P} = \text{BPP}$ [5] and $\text{ALMOST-NP} = \text{AM}$ [26]. The condition $\text{AM} \cap \text{coAM} \neq \text{BPP}$ implies that there exist problems in $\text{ALMOST-NP} \cap \text{ALMOST-coNP}$ that are not in ALMOST-P . Since the intersection of measure 1 classes is measure 1, this implies $\text{NP}^A \cap \text{coNP}^A \neq \text{P}^A$ relative to a random oracle A with probability 1. Recent work of Ghosal et al. [10] shows that if $\text{UP} \not\subseteq \text{RP}$, then $\text{P}^A \neq \text{NP}^A \cap \text{coNP}^A$ with probability 1 for a random oracle A . In Section 7 we pivot from permutation randomness to classical random oracles and show that resolving the long-standing question “does $\text{P}^R = \text{NP}^R \cap \text{coNP}^R$ with probability 1?” is tightly linked to quantitative structure inside EXP. Leveraging the conditional oracle separations of Tardos [29] and of Ghosal et al. [10], we prove that if $\text{P}^R = \text{NP}^R \cap \text{coNP}^R$ holds almost surely, then several familiar subclasses of EXP obey strong 0-1 laws: specifically, either $\text{NP} \cap \text{coNP}$, $\text{UP} \cap \text{coUP}$, (and, in a weaker form, UP vs. FewP) each has $\mathbf{p}$ -measure 0 or else fills all of EXP. Consequently, non-measurability of any one of these classes immediately forces $\text{P}^R \neq \text{NP}^R \cap \text{coNP}^R$ with probability 1. We further show that placing the same oracle separation in $\mathbf{p}_2$ measure would collapse BPP below EXP, thereby framing the random-oracle problem in terms of concrete measure-theoretic thresholds inside exponential time.

1.5 Organization

This paper is organized as follows: Section 2 contains preliminaries. Section 3 develops permutation martingales, resource-bounded permutation measure, and random permutations. Elementary properties of $\mathbf{p}$ -random permutations are presented in Section 4. In Section 5, we prove our main results on random permutations for P vs. $\text{NP} \cap \text{coNP}$. Section 6 contains our results on $\text{NP} \cap \text{coNP}$ versus quantum computation relative to a random permutation. In Section 7 we present our results on random oracles and 0-1 laws. We conclude in Section 8 with some open questions.

2 Preliminaries

The binary alphabet is $\Sigma = \{0, 1\}$, the set of all binary strings is Σ^* , the set of all binary strings of length n is Σ^n , and the set of all infinite binary sequences is Σ^∞ . The empty string is denoted by λ . We use the standard enumeration of strings, $s_0 = \lambda, s_1 = 0, s_2 = 1, s_3 = 00, s_4 = 01, \dots$. The characteristic sequence of a language A is the sequence $\chi_A \in \Sigma^\infty$, where $\chi_A[n] = 1 \iff s_n \in A$. We refer to $\chi_A[s_n] = \chi_A[n]$ as the characteristic bit of s_n in A . A language A can alternatively be seen as a subset of Σ^* , or as an element of Σ^∞ via identification with its characteristic sequence χ_A . Given strings x, y we denote by $[x, y]$ the set of all strings z such that $x \leq z \leq y$. For any string s_n and natural number k , $s_n + k$ is the string s_{n+k} ; e.g. $\lambda + 4 = 01$. Similarly we denote by $A[x, y]$ the substring of the characteristic sequence χ_A that corresponds to the characteristic bits of the strings in $[x, y]$. We use parentheses for intervals that do not include the endpoints. We write $A \upharpoonright n$ for the length n prefix of A . A statement $\mathcal{S}_n$ holds infinitely often (written i.o.) if it holds for infinitely many n , and it holds almost everywhere (written a.e.) if it holds for all but finitely many n .

3 Permutation Martingales and Permutation Measure

3.1 Permutation Measure Space

Resource-bounded measure is typically defined in the Cantor Space $\mathbb{C} = \{0, 1\}^\infty = 2^\mathbb{N}$ of all infinite binary sequences. For measure in $\mathbb{C}$, we use the open balls or cylinders $\mathbb{C}_w = w \cdot \mathbb{C}$ that have measure $\mu(\mathbb{C}_w) = 2^{-|w|}$ for each $w \in \Sigma^*$. Let $\mathcal{C}$ be the σ -algebra generated by $\{\mathbb{C}_w \mid w \in \{0, 1\}^*\}$. Resource-bounded measure and algorithmic randomness typically work in the probability space $(\mathbb{C}, \mathcal{C}, \mu)$.

We only consider permutations in Π , the set of permutations on $\{0, 1\}^*$ that preserve string lengths. Given a permutation $\pi \in \Pi$, we denote by π_n the permutation π restricted to $\{0, 1\}^n$ i.e., π_n is a permutation on $\{0, 1\}^n$. Similarly, Π_n denotes the set of permutations in Π restricted to $\{0, 1\}^n$. Bennett and Gill [5] considered random permutations by placing the uniform measure on each Π_n and taking the product measure to get a measure on Π . We now define this measure space more formally so we may place martingales on it.

Standard resource-bounded measure identifies a language $A \subseteq \{0, 1\}^*$ with its infinite binary characteristic sequence $\chi_A \in \mathbb{C}$. For permutations, we analogously use the value sequence consisting of all function values.

Definition 3.1. The *value sequence* of a permutation $f \in \Pi$ is the sequence

$$\nu_f = [f(s_0), f(s_1), f(s_2), \dots]$$

of function values where $s_0, s_1, s_2, \dots$ is the standard enumeration of $\{0, 1\}^*$.

We identify a permutation $f \in \Pi$ with its value sequence ν_f . Initial segments of permutations are called prefix partial permutations.

Definition 3.2. A *prefix partial permutation* is a list $g = [g(s_0), \dots, g(s_{N-1})]$ of function values for some $N \geq 0$ where no value is repeated and $|g(s_i)| = |s_i|$ for all $0 \leq i < N$. We let $\text{PP}\Pi$ denote the class of all prefix partial permutations.

We write each $g \in \text{PP}\Pi$ as a list $g = [g(s_0), \dots, g(s_{N-1})]$. The *length* of g is N , the number of function values assigned, and is denoted $|g|$. We use $[]$ to denote the *empty list*, the list of length 0. We write $f \upharpoonright N$ for the length N prefix partial permutation of $f \in \Pi$.

Definition 3.3. For each $g = [g(s_0), \dots, g(s_{N-1})] \in \text{PP}\Pi$, the *cylinder* of all permutations in Π that extend g is

$$[g] = \{h \in \Pi \mid h(s_0) = g(s_0), \dots, h(s_{N-1}) = g(s_{N-1})\}.$$

For measure in Π , we are taking the uniform distribution on the set of all Π_n of length-preserving permutations for all n . Our basic open sets are $\{[g] \mid g \in \text{PP}\Pi\}$. Suppose $g \in \text{PP}\Pi$ has $|g| = 2^n - 1$ for some $n \geq 0$. Then, following Bennett and Gill [5], the measure

$$\mu([g]) = \prod_{k=0}^{n-1} \frac{1}{(2^k)!}$$

is easy to define because the distribution is uniform over the $(2^k)!$ permutations at each length. If $2^n - 1 \leq |g| < 2^{n+1} - 1$, let $m = |g| - 2^n + 1$ and then

$$\mu([g]) = \left(\prod_{k=0}^{n-1} \frac{1}{(2^k)!} \right) \frac{(2^n - m)!}{(2^n)!} = \left(\prod_{k=0}^{n-1} \frac{1}{(2^k)!} \right) \frac{1}{P(2^n, m)},$$

where $P(n, k) = \frac{n!}{(n-k)!}$ denotes the number of k -permutations on n elements. For convenience, we commonly write $\mu(g) = \mu(\llbracket g \rrbracket)$.

Let $\mathcal{F}_\Pi = \sigma(\text{PP}\Pi)$ be the σ -algebra generated by the collection of all $\llbracket g \rrbracket$ where $g \in \text{PP}\Pi$. By Carathéodory's extension theorem, μ extends uniquely to $\mathcal{F}_\Pi$, yielding the probability space $(\Pi, \mathcal{F}_\Pi, \mu)$. We will work in this probability space. Because μ is outer regular, we have the typical open cover characterization of measure zero:

Theorem 3.4. *A class $X \subseteq \Pi$ has measure 0 if and only if for every $\epsilon > 0$, there is an open covering $G = \{g_0, g_1, \dots\} \subseteq \text{PP}\Pi$ such that*

$$\sum_{i=0}^{\infty} \mu(g_i) < \epsilon \quad \text{and} \quad X \subseteq \bigcup_{i=0}^{\infty} \llbracket g_i \rrbracket.$$

3.2 Permutation Martingales

In resource-bounded measure in Cantor Space, a *martingale* is a function $d : \Sigma^* \rightarrow [0, \infty)$ such that for all $w \in \Sigma^*$, we have the following averaging condition:

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

A martingale in Cantor space may be viewed as betting on the membership of strings in a language. The standard enumeration of $\{0, 1\}^*$ is $s_0 = \lambda, s_1 = 0, s_2 = 1, s_3 = 00, s_4 = 01, \dots$. In the i^{th} stage of the game, the martingale has seen the membership of the first i strings and bets on the membership of s_i in the language. The martingale's value is updated based on the outcome of the bet. For further background on resource-bounded measure, we refer to [1, 7, 11, 20, 21].

A permutation martingale operates similarly, but instead of betting on the membership of a string in a language it bets on the next function value of the permutation. Instead of seeing the characteristic string of a language, a permutation martingale sees a *prefix partial permutation*, which is a list of permutation function values $g = [g(s_0), \dots, g(s_{i-1})]$ satisfying $|g(s_i)| = |s_i|$ for all i . The permutation martingale will bet on the next function value $g(s_i)$. The current *betting length* is the length of the next string $s_{|g|}$ in the standard enumeration: $l(g) = |s_{|g|}|$. The set of *free strings* available for the next function value is

$$\text{free}(g) = \{x \in \{0, 1\}^{l(g)} \mid x \text{ is not in } g\}.$$

For example, $\text{free}([\lambda]) = \{0, 1\}$, $\text{free}([\lambda, 1, 0, 11]) = \{00, 01, 10\}$, and $\text{free}([\lambda, 1, 0, 11, 00, 01]) = \{10\}$.

We now introduce our main conceptual contribution, permutation martingales.

Definition 3.5. A *permutation martingale* is a function $d : \text{PP}\Pi \rightarrow [0, \infty)$ such that for every prefix partial permutation $g \in \text{PP}\Pi$,

$$d(g) = \frac{1}{|\text{free}(g)|} \sum_{x \in \text{free}(g)} d(g, x),$$

where (g, x) is the result of appending x to g .

Success is defined for permutation martingales analogously to success for classical martingales.

Definition 3.6. Let d be a permutation martingale. We say d *succeeds on* $f \in \Pi$ if

$$\limsup_{N \rightarrow \infty} d(f \upharpoonright N) = \infty.$$

The *success set* of d is

$$S^\infty[d] = \{f \in \Pi \mid d \text{ succeeds on } f\}$$

and the *unitary success set* of d is the set

$$S^1[d] = \{f \in \Pi \mid (\exists n) d(f \upharpoonright n) \geq 1\}.$$

In the remainder of this section, we establish the analogue of Ville's theorem [31] for measure in Π and permutation martingales.

Theorem 3.7. *The following statements are equivalent for every $X \subseteq \Pi$:*

1. X has measure 0.
2. For every $\epsilon > 0$, there is a permutation martingale d with $d(\lambda) < \epsilon$ and $X \subseteq S^1[d]$.
3. There is a permutation martingale d with $X \subseteq S^\infty[d]$.

First, we need a few lemmas.

Lemma 3.8. *If $g \in \text{PP}\Pi$, then there is a permutation martingale d_g with $d_g(\lambda) = \mu(g)$ and $S^1(d_g) = \llbracket g \rrbracket$.*

Proof. Let $N = |g|$ and define $d_g(x) = \Pr_{|h|=N} [g \sqsubseteq x \sqsubseteq h]$, where we choose $h \in \text{PP}\Pi$ of length N uniformly at random. $\square$

A *premeasure* on Π is a function $\rho : \text{PP}\Pi \rightarrow [0, 1]$ such that $\rho(\lambda) = 1$ and for all $g \in \text{PP}\Pi$,

$$\rho(g) = \sum_{w \in \text{free}(g)} \rho(g, w).$$

A *prefix set* in $\text{PP}\Pi$ is a set $W \subseteq \text{PP}\Pi$ such that no element of W is a prefix of any other element.

Lemma 3.9. *If $W \subseteq \text{PP}\Pi$ is a prefix-free set and ρ is any premeasure on Π , then*

$$\sum_{g \in W} \rho(g) \leq 1.$$

Proof. Because W provides a disjoint collection of cylinders, the total measure of the cylinders is at most 1. (A rigorous proof may be given using induction.) $\square$

Lemma 3.10. *If W is a prefix-free set and d is any permutation martingale, then*

$$\sum_{g \in W} \mu(g) d(g) \leq d(\llbracket \cdot \rrbracket).$$

Proof. The function $\mu(g) d(g)$ is a premeasure on Π , so this follows from Lemma 3.9 $\square$

For $k \geq 1$, let $S^k[d] = \{\pi \in \Pi \mid (\exists n) d(\pi \upharpoonright n) \geq k\}$.

Lemma 3.11. *For any permutation martingale and $k \geq 1$,*

$$\mu(S^k[d]) \leq \frac{d([\])}{k}.$$

and there is an open cover W_k with $\mu(W_k) = \mu(S^k[d])$.

Proof. Let d be a permutation martingale. Define for each $k \geq 1$,

$$W_k = \{g \in \text{PP}\Pi \mid d(g) \geq k \text{ and } d(h) < k \text{ for all proper prefixes } h \text{ of } g\}.$$

Then

$$S^k[d] = \bigcup_{g \in W_k} \llbracket g \rrbracket,$$

so $\mu(W_k) = \mu(S^k[d])$. By Lemma 3.10,

$$d([\]) \geq \sum_{g \in W_k} \mu(g)d(g) \geq \sum_{g \in W_k} \mu(g)k = k \sum_{g \in W_k} \mu(g) = k\mu(W_k).$$

□

We are now ready to prove Theorem 3.7.

Proof of Theorem 3.7. Suppose 1 is true. Let G be a covering of X with $\mu(G) < \epsilon$. Then define d by $d = \sum_{g \in G} d_g$ where each d_g comes from Lemma 3.8. We have $d(\lambda) = \mu(G) < \epsilon$ and $X \subseteq S^1[d]$.

Suppose 2 is true. For each $k \in \mathbb{N}$, let d_k be a martingale with $d_k(\lambda) < 2^{-k}$ and $X \subseteq S^1[d_k]$. Without loss of generality, we assume that if $d_k(g) \geq 1$, then $d_k(h) = d_k(g)$ for all $g \sqsubseteq h$. Define d by $d = \sum_{k \in \mathbb{N}} d_k$. Let $A \in X$. For every k , there exists n_k such that $d_k(A \upharpoonright n_k) \geq 1$. Let $m_k = \max\{n_1, \dots, n_k\}$. Then $d(A \upharpoonright m_k) \geq k$. Since k is arbitrary, $A \in S^\infty[d]$. Therefore $X \subseteq S^\infty[d]$.

Suppose 3 is true. Let d be a martingale with $d(\lambda) = 1$ and $X \subseteq S^\infty[d]$. Let $\epsilon > 0$ and let $k \geq 1$ such that $\frac{1}{k} < \epsilon$. Let B_k be the set of all shortest g with $d(g) \geq k$. Then $X \subseteq B_k$, B_k is an open set, and $\mu(B_k) \leq \frac{1}{k} < \epsilon$ by Lemma 3.11. □

3.3 A Permutation Martingale Example

We construct a permutation martingale d that succeeds on any length-preserving permutation whose restriction to length n is a cycle permutation for all but finitely many n . We partition the initial capital into infinitely many shares $a_i = 1/i^2$. For each i , the share a_i is used to bet on the event that, for all $n \geq i$, the length- n restriction of the permutation is a cycle permutation.

The betting strategy is simple: when moving from length $n - 1$ to n , the martingale wagers all relevant capital on the image of the n -bit string $1^{n-1}0$. In the final step of forming a cycle of length 2^n , there are exactly two choices for the image of $1^{n-1}0$. One choice yields a cycle of length 2^n ; the other does not. Since it is a binary choice, the martingale places its entire stake a_i (for all $i \leq n$) on the cycle outcome, thereby doubling its capital whenever the cycle is formed.

Hence, on any permutation whose restriction to length n is a cycle permutation for all but finitely many n , infinitely many of these bets succeed. Consequently, each of those corresponding shares a_i grows without bound, and so the overall martingale d succeeds on all such permutations.

3.4 Permutation Martingales as Random Variables

Hitchcock and Lutz [13] showed how the martingales used in computational complexity are a special case of martingales used in probability theory. We explain how this extends to permutation martingales. Given a martingale $d : \{0, 1\}^* \rightarrow [0, \infty)$, Hitchcock and Lutz define the random variable $\xi_{d,n} : \mathbb{C} \rightarrow [0, \infty)$ by $\xi_{d,n}(S) = d(S \upharpoonright n)$ for each $n \geq 0$. Let $\mathcal{M}_n = \sigma(\{C_w \mid w \in \{0, 1\}^n\})$ be the σ -algebra generated by the cylinders of length n . Then the sequence $(\xi_{d,n} \mid n \geq 0)$ is a martingale in the probability theory sense with respect to the filtration $(\mathcal{M}_n \mid n \geq 0)$: for all $n \geq 0$, $E[\xi_{d,n+1} \mid \mathcal{M}_n] = \xi_{d,n}$.

Similarly, given a permutation martingale $d : \text{PP}\Pi \rightarrow [0, \infty)$, for each N we can define the random variable $X_{d,N} : \Pi \rightarrow [0, \infty)$ by $X_{d,N}(f) = d(f \upharpoonright N)$ for each $N \geq 0$. Let

$$\mathcal{G}_N = \sigma(\{\llbracket g \rrbracket \mid g \in \text{PP}\Pi \text{ and } |g| = N\})$$

be the σ -algebra generated by the cylinders in $\text{PP}\Pi$ of length N . Then $(X_{d,N} \mid N \geq 0)$ is a martingale in the probability theory sense with respect to the filtration $(\mathcal{G}_N \mid N \geq 0)$: for all $N \geq 0$, $E[X_{d,N+1} \mid \mathcal{G}_N] = X_{d,N}$.

3.5 Resource-Bounded Permutation Measure

We follow the standard notion of computability for real-valued functions [20] to define resource-bounded permutation martingales.

Definition 3.12. Let $d : \text{PP}\Pi \rightarrow [0, \infty)$ be a permutation martingale.

1. d is *computable in time* $t(n)$ if there is an exactly computable $\hat{d} : \text{PP}\Pi \times \mathbb{N} \rightarrow \mathbb{Q}$ such that for all $f \in \text{PP}\Pi$ and $r \in \mathbb{N}$, $|d(f) - \hat{d}(f, r)| \leq 2^{-r}$ and $\hat{d}(f, r)$ is computable in time $t(|f| + r)$.
2. d is *computable in space* $s(n)$ if there is an exactly computable $\hat{d} : \text{PP}\Pi \times \mathbb{N} \rightarrow \mathbb{Q}$ such that for all $f \in \text{PP}\Pi$ and $r \in \mathbb{N}$, $|d(f) - \hat{d}(f, r)| \leq 2^{-r}$ and $\hat{d}(f, r)$ is computable in space $s(|f| + r)$.
3. If d is computable in polynomial time, then d is a *p-permutation martingale*.
4. If d is computable in quasipolynomial time, then d is a *p₂-permutation martingale*.
5. If d is computable in polynomial space, then d is a *pspace-permutation martingale*.
6. If d is computable in quasipolynomial space, then d is a *p₂space-permutation martingale*.

We are now ready to define resource-bounded permutation measure.

Definition 3.13. Let $\Delta \in \{\text{p}, \text{p}_2, \text{pspace}, \text{p}_2\text{space}\}$. Let $X \subseteq \Pi$ and $X^c = \Pi - X$ be the complement of X within Π .

1. X has Δ -measure 0, written $\mu_\Delta(X) = 0$, if there is a Δ -computable permutation martingale d with $X \subseteq S^\infty[d]$.
2. X has Δ -measure 1, written $\mu_\Delta(X) = 1$, if $\mu_\Delta(X^c) = 0$

Definition 3.14. Let $\Delta \in \{\text{p}, \text{p}_2, \text{pspace}, \text{p}_2\text{space}\}$. A permutation $\pi \in \Pi$ is Δ -random if π is not contained in any Δ -measure 0 set.

Equivalently, π is Δ -random if no Δ -martingale succeeds on π .

3.6 Permutation Betting Games

Originated in the field of algorithmic information theory, betting games are a generalization of martingales [24, 25], which were introduced to computational complexity by Buhrman et al. [7]. Similar to martingales, betting games can be thought of as strategies for betting on a binary sequence, except that with betting games we have the additional capability of selecting which position in a sequence to bet on next. In other words, a betting game is permitted to select strings in a nonmonotone order, with the important restriction that it may not bet on the same string more than once (see Buhrman et al. [7] for more details).

A permutation betting game is a generalization of a permutation martingale, implemented by an oracle Turing machine, where it is allowed to select strings in nonmonotone order. Prefixes of permutation betting games can be represented as *ordered partial permutations* defined below.

Definition 3.15. An *ordered partial permutation* is a list $g = [(x_1, y_1), \dots, (x_n, y_n)]$ of pairs of strings for some $n \geq 0$ where for all $1 \leq i < j \leq n$, $x_i \neq x_j$ and $y_i \neq y_j$, and $|x_i| = |y_i|$ for all $1 \leq i \leq n$. We let $\text{OP}\Pi$ denote the class of all ordered partial permutations.

For a permutation betting game, the averaging condition takes into consideration the length of the next string to be queried as follows. Let $w \in \text{OP}\Pi$ be the list of queried strings paired with their images, and $a \in \{0, 1\}^n$ be the next string the betting game will query. Define $\text{free}(w, n)$ to be the set of length- n strings that are available for the next function value, i.e., length- n strings that are not the function value of any of the queried strings. Then the following averaging condition over free strings of length n must hold for the permutation betting game $d : \text{OP}\Pi \rightarrow [0, \infty)$

$$d(w) = \sum_{b \in \text{free}(w, n)} \frac{d(w[a, b])}{|\text{free}(w, n)|}$$

where $w[a, b]$ is the list w appended with the pair (a, b) .

Definition 3.16. A betting game is a $t(n)$ -time betting game if for all n , all strings of length n have been queried by time $t(2^n)$.

We define betting game measure 0 and betting game randomness analogously.

Definition 3.17. Let $\Delta \in \{\text{p}, \text{p}_2, \text{pspace}, \text{p}_2\text{space}\}$.

1. A class $X \subseteq \Pi$ has Δ -betting-game measure 0 if there is a Δ -computable permutation betting game d with $X \subseteq S^\infty[d]$.
2. A permutation $\pi \in \Pi$ is Δ -betting game random if no Δ -betting game succeeds on π .

3.7 Measure Conservation

Lutz's Measure Conservation Theorem implies that resource-bounded measure gives nontrivial notions of measure within exponential-time complexity classes: $\mu_{\text{p}}(\text{E}) \neq 0$ and $\mu_{\text{p}_2}(\text{EXP}) \neq 0$. Let PermE be the class of length-preserving permutations that can be computed in $2^{O(n)}$ time and PermEXP be the class of length-preserving permutations that can be computed in $2^{n^{O(1)}}$ time. We show that our notions of permutation measure have conservation theorems within these classes of exponential-time computable permutations.

Lemma 3.18. For any $t(2^n)$ -time permutation martingale D , we can construct a $O(2^{2n}t(2^n + 2^{2n}))$ -time permutation that is not succeeded on by D .

Proof. Let Q be a Turing machine that operates as follows on input $x \in \{0, 1\}^n$. The machine Q simulates the permutation martingale D on prefix partial permutations starting from the empty list and ending when the image of x is added to the list. During the simulation, whenever D bets on a string y , Q maps y to the first string to which no other string has been mapped, and for which D 's capital decreases the most, up to an additive error of 2^{-2n} . After this simulation ends, D outputs the string assigned to x by this simulation. Note that when computing $Q(x)$, we first compute $Q(y)$ for all $y < x$.

Clearly, Q computes a permutation on which D 's capital never exceeds its initial value plus

$$\sum_{n=1}^{\infty} \sum_{x \in \{0,1\}^n} 2^{-2n} = \sum_{n=1}^{\infty} 2^{-n} = 1,$$

so D cannot succeed on this permutation. The runtime of Q on input x is $O(2^{2n}t(2^n + 2^{2n}))$, since finding the image that minimizes D 's capital on a length- n string x requires scanning $O(2^n)$ candidates for each of the $O(2^n)$ strings preceding x , and each evaluation of the martingale's value (to within an additive error of 2^{-2n}) takes $t(2^n + 2^{2n})$ time. $\square$

The following theorem follows from Lemma 3.18.

Theorem 3.19. 1. PermE does not have $\mathbf{p}$ -permutation measure 0.

2. PermEXP does not have $\mathbf{p}_2$ -permutation measure 0.

Proving similar results for betting games turns out to be more challenging, given that they are allowed to bet on strings in an adaptive order. To address this, we define the following class of *honest* betting games.

Definition 3.20. A $\log(t(2^n))$ -honest $t(n)$ -permutation betting game is a $t(n)$ -time permutation betting game such that for all languages A , for all n , all non-zero bets by time $t(2^n)$ are for strings of length at most $\log(t(2^n))$.

We use this definition in the following lemma.

Lemma 3.21. For any $\log(t(2^n))$ -honest $t(2^n)$ -permutation betting game G , we can construct a $O(t(2^n)^2 t(t(2^n) + t(2^n)^2))$ -time permutation that is not succeeded on by G .

Proof. This proof is similar to the proof of Lemma 3.18. So we focus on the main difference between the proofs.

We construct a $\mathbf{DTIME}(t(2^n)^2 t(t(2^n) + t(2^n)^2))$ permutation using the same strategy as was used in the previous proof. The only difference is when the simulated betting game G queries a string of length greater than $m = \log(t(2^n))$. In this case, the betting game answers the query with the first string of length m that has not yet been made an image. This step takes $O(t(2^n))$ time. Since the betting game does not bet on this string, its capital is unaffected. When the betting game queries strings of length at most m , it behaves just like it did in the previous proof, it finds a string that minimizes the betting game's capital within an additive factor of 2^{-2n} .

It is easy to see that the betting game does not succeed on this permutation. This permutation can be computed in $O(t(2^n)^2 t(t(2^n) + t(2^n)^2))$. This is because on input $x \in \{0, 1\}^n$, the simulation queries at most $t(2^n)$ strings, for each queried string it examines at most $t(2^n)$ strings to be its image, and it takes $O(t(2^{\log(t(2^n))}) + 2^{2\log(t(2^n))}) = O(t(t(2^n) + t(2^n)^2))$ time to evaluate the betting on each image within the appropriate margin of error. $\square$

Theorem 3.22. 1. PermE does not have $O(n)$ -honest $\mathbf{p}$ -permutation betting game measure 0.
2. PermEXP does not have $O(n^k)$ -honest $\mathbf{p}_2$ -permutation betting game measure 0.

Since $\mathbf{pspace}$ -permutation martingales can simulate $O(n)$ -honest $\mathbf{p}$ -betting-games, and $\mathbf{p}_2\mathbf{space}$ -permutation martingales can simulate $O(n^k)$ -honest $\mathbf{p}_2$ -betting-games, we have the following:

Proposition 3.23. Let π be a permutation.

1. If π is a $\mathbf{pspace}$ -random permutation, then π is $O(n)$ -honest $\mathbf{p}$ -betting game random.
2. If π is a $\mathbf{p}_2\mathbf{space}$ -random permutation, then π is $O(n^k)$ -honest $\mathbf{p}_2$ -betting game random.

4 Elementary Properties of Random Permutations

In this section, we explore fundamental properties of random permutations that provide insights into how permutation martingales and betting games operate. Understanding these properties is crucial for applying permutation randomness in computational complexity. We show that random permutations are computationally difficult to compute and to invert. We then investigate the relationship between random permutations and random oracles, showing how random permutations can generate random oracles.

4.1 Intractability of Random Permutations

Definition 4.1. A permutation $\pi \in \Pi$ is *noticeably polynomial time* if there are polynomials p, q and TM M such that for infinitely many n , M computes π on at least $2^n/p(n)$ strings of length n in $q(n)$ time for each string.

Theorem 4.2. The set $X = \{\pi \in \Pi \mid \pi_n \text{ is noticeably polynomial time}\}$ has $\mathbf{p}$ -permutation measure 0.

The proof uses a simple averaging argument: we partition the set of length- n strings into $2^n/n^{\lg n}$ subintervals, each of size $n^{\lg n}$. By the noticeably-polynomial-time property of the permutations in X , at least one subinterval contains superpolynomially many strings whose images are computable. The martingale then identifies a sufficiently small subset of these strings and makes correct predictions often enough to succeed.

Proof of Theorem 4.2. We design a polynomial-time permutation-martingale game that succeeds on X . The martingale succeeds on X by going through all polynomial time TMs and finding a size- $n^{1/2}$ subset of $\{0, 1\}^n$ where it wins enough bets to succeed.

This martingale operates in stages. In stage n , the martingale uses the first n polynomial TMs to bet on $\{0, 1\}^n$. We partition $\{0, 1\}^n$ into $2/n^{\lg n}$ subsets of size $n^{\lg n}$. For each partition, we consider all its size- $n^{1/2}$ subsets. Then we run the n TMs on each size- $n^{1/2}$ subset of each partition. So, in total we consider $N_n = (2^n/n^{\lg n}) \binom{n^{\lg n}}{n^{1/2}} = 2^{\Theta(n)}$ subsets. We use the TMs outputs to predict the permutation we are betting on. Each size- $n^{1/2}$ subset and TM is given capital $\frac{1}{n^2 N_n}$ to make its bet. Next, we show that for any $\pi \in X$, for infinitely many n , there is a subset that increases its capital to $\omega(1)$.

We now specify how the martingale bets in stage n . First, we naturally partition $\{0, 1\}^n$ into $2^n/n^{\lg n}$ contiguous subsets $T_i = [0^n + i n^{\lg n}, 0^n + (i+1)n^{\lg n})$, for $i \in [0, 2^n/n^{\lg n})$. For each T_i , the martingale examines all its size- $n^{1/2}$ subsets. It then runs each size- $n^{1/2}$ subset on the first n TMs,

M_k for $k \in [1, n]$. Let $M_{i,j,k}$ denote the image of the j th size- $n^{1/2}$ subset of T_i under the function computed by M_k . As mentioned before, we use $\frac{1}{n^2 N_n}$ to bet with each $M_{i,j,k}$. The martingale uses $\frac{1}{n^2 N_n}$ to bet that each $M_{i,j,k}$ is correct, if $M_{i,j,k}$ is incorrect, the martingale loses all its capital reserved for betting on $M_{i,j,k}$, otherwise, its capital increases by a factor described later.

We now argue that for each $\pi \in X$, there are infinitely many n , such that the capital of the martingale after betting on $\{0, 1\}^n$ is $\omega(1)$. Suppose that π_n is computable by at least one of the first n polynomial time TMs on a noticeable subset of $\{0, 1\}^n$ i.e., π_n is computed on a subset of size at least $2^n/n^k$ for some positive constant k . Let us call such a TM, a good TM. Using a simple averaging argument, we can see that there must be some size- $n^{\lg n}$ subset T_i where a good TM computes π_n correctly on a subset of size at least $n^{\lg n}/n^k$. For sufficiently large n , $n^{\lg n-k} \geq n^{1/2}$. Therefore, for sufficiently large n , we can always find size- $n^{1/2}$ subsets where a good TM computes π . In fact, for sufficiently large n , there are $\Omega(2^n/n^{\lg n+k})$ subsets, T_i , with at least $n^{1/2}$ strings that are computed by a good TM.

We now examine how much capital the martingale gains after betting with a good TM. A correct prediction on a string $s \in T_i = [0^n + i n^{\lg n}, 0^n + (i+1)n^{\lg n})$ increases the amount bet on s by a factor of at least $2^n - (i+1)n^{\lg n} + 1$ for any $i \in [0, 2^n/n^{\lg n+k})$. In the worst case, some T_i with $i = O(2^n/n^{\lg n+k})$ will have $n^{1/2}$ strings that are correctly predicted by a good TM. Therefore the capital wagered on a the good TM increases by a factor of at least

$$(2^n - (i+1)n^{\lg n} + 1)^{n^{1/2}} = 2^{n^{3/2}}(1 - (i+1)2^{-n}n^{\lg n} + 2^{-n})^{n^{1/2}} = \omega(2^{n^{1.4}}).$$

The last equality follows because $k > 0$ and $i = O(2^n/n^{\lg n+k})$.

Since this value grows much faster than $\frac{1}{n^2 N_n} = \omega(2^{-1.1n})$, the value reserved for betting on each $M_{i,j,k}$, we see that our martingale succeeds on X .

Finally, we argue that our martingale operates in linear exponential time. The most time consuming step of the martingale is the simulation of n polynomial time TMs that are run on $N_n = 2^{\Theta(n)}$ subsets with $n^{1/2}$ length- n strings. Clearly this can be done in $2^{\Theta(n)}$ time. $\square$

Corollary 4.3. *If $\pi \in \Pi$ is $\mathbf{p}$ -random, then any polynomial-time TM will be able to compute π on at most a $1/\text{poly}$ fraction for all sufficiently large n .*

Similarly, we can show that random permutations are hard to invert on a noticeable subset infinitely often. The main difference is that we search for TMs inverting the permutation rather than TMs that compute the permutation.

Definition 4.4. A permutation $\pi \in \Pi$ is *noticeably invertible* if there is a polynomial-time TM M and a polynomial p such that for infinitely many n , $|\{x \in \{0, 1\}^n \mid M(\pi(x)) = x\}| \geq 2^n/p(n)$.

Theorem 4.5. *The set $X = \{\pi \in \Pi \mid \pi_n \text{ is noticeably invertible}\}$ has $\mathbf{p}$ -permutation measure 0.*

Proof. This proof closely mirrors the argument for Theorem 4.2. The key difference is that we treat the outputs of the TMs as *preimages* rather than images of the permutation. Specifically, given a size- $n^{1/2}$ subset $S = \{x_1, \dots, x_{n^{1/2}}\} \subseteq \{0, 1\}^n$, we run each TM M on x_i and interpret the output $M(x_i)$ (if it is length- n) as a prediction that $\pi(M(x_i)) = x_i$.

The martingale allocates capital to each subset-TM pair and places bets accordingly. It only bets when the outputs $M(x_i)$ are all distinct and of length n . As in Theorem 4.2, noticeable invertibility ensures that for infinitely many n , there exists a TM making sufficiently many correct inversion predictions. In such cases, the martingale's capital grows superpolynomially, and hence it succeeds. $\square$

4.2 Random Permutations versus Random Oracles

Bennett and Gill used random permutations, rather than random languages, to separate $\mathbf{P}$ from $\mathbf{NP} \cap \mathbf{coNP}$. It is still unknown whether random oracles separate $\mathbf{P}$ from $\mathbf{NP} \cap \mathbf{coNP}$. In this section, we examine how random permutations yield random languages. We show that a $\mathbf{p}$ -random permutation can be used to generate a $\mathbf{p}$ -random language. All of the results in this section are stated for $\mathbf{p}$ -randomness. They also hold for $\mathbf{p}_2$ -randomness.

Given a permutation $\pi \in \Pi$, we define the language

$$L_\pi = \{x \mid \text{the first bit of } \pi(0^{2|x}|x) \text{ is } 1\}.$$

For a set of permutations $X \subseteq \Pi$, we define the set of languages

$$L_X = \{L_\pi \mid \pi \in X\}.$$

Lemma 4.6. *For any set of permutations $X \subseteq \Pi$, if a $\mathbf{p}$ -computable martingale d succeeds on the set of languages $L_X = \{L_\pi \mid \pi \in X\}$, then there is a $\mathbf{p}$ -computable permutation martingale that succeeds on X .*

Proof. We will use d to design a $\mathbf{p}$ -computable permutation martingale D , that succeeds on any $\pi \in X$.

The main idea behind D 's design is to use the capital of the martingale d to bet on the images of strings that affect the membership of strings in L_π . The bet placed by D on each possible image of the string being bet on is proportional to the bet placed by d on the membership in L_π of the string whose membership is determined by the image. For example, the bet placed by D on the image of $0^{2|x}|x$ being $1y$ is proportional to the bet placed by d on x being in L_π .

The permutation martingale D only bets on strings of form $0^{2n}x$, where $x \in \{0,1\}^n$ and $n > 0$ (this is the string that determines the membership of x in L_π). Given a string w , let $\pi \upharpoonright w$ be the prefix list of ν_π that consists of the images of all the strings less than w . We let $\text{free}(\pi \upharpoonright w)$ be the (nonempty) set of length- n strings that have not yet appeared in the image of the prefix partial permutation $\pi \upharpoonright w$. Where n is the smallest integer for which there exists strings of length n that have not been mapped to any string. For $b \in \{0,1\}$, $\text{free}(\pi \upharpoonright w)_b$ is the set of strings of $\text{free}(\pi \upharpoonright w)$ that have b as their first bit. Let $F(\pi \upharpoonright w)$ and $F(\pi \upharpoonright w)_b$ denote the size of $\text{free}(\pi \upharpoonright w)$ and $\text{free}(\pi \upharpoonright w)_b$, respectively. Finally, let $L(\pi \upharpoonright w)$ be a prefix of the characteristic sequence of L_π , it includes the characteristic bits for all strings whose membership can be determined from $\pi \upharpoonright w$. For example, if $w = 0^{3n+1}$, then $L(\pi \upharpoonright w)$ is the characteristic prefix for the subset L_π that consists of all string of length at most n .

We now define D . As mentioned previously, D only bets on strings that have the form $0^{2n}x$, where $x \in \{0,1\}^n$ and $n > 0$. These are the strings that determine the membership of strings in L_π . The martingale D starts with initial capital $d(\lambda)$, the initial capital of the martingale that succeeds on L_π . Without loss of generality, we assume that d is non-zero. When betting on the images of string $w = 0^{2|x}|x$ with $x \in \{0,1\}^n$, for each length- $3n$ string beginning with bit b , the martingale wagers $\frac{d(L(\pi \upharpoonright w)_b)}{2d(L(\pi \upharpoonright w))F(\pi \upharpoonright w)_b}$ fraction of its current capital. The term $\frac{d(L(\pi \upharpoonright w)_b)}{2d(L(\pi \upharpoonright w))}$ is the fraction of d 's current capital used to bet on the characteristic bit of x in L_π being b . We then divide this term by $F(\pi \upharpoonright w)_b$, the number of length- $3n$ strings that make the characteristic bit of x in L_π to be b .

It is easy to see that D is computable in $\mathbf{p}$ if d is also computable in $\mathbf{p}$. Given $\pi \upharpoonright w$, it takes linear time to compute the string $L(\pi \upharpoonright w)$ and the number $F(\pi \upharpoonright w)_b$. It takes $\mathbf{p}$ -time to compute $d(L(\pi \upharpoonright w)_b)$ and $2d(L(\pi \upharpoonright w))$. All that is left after that is to perform a multiplication and a division, both of which requires $\mathbf{p}$ -time.

We now argue that D succeeds on L_π . Let us examine the martingale's capital after betting on string $w = 0^{2|x|}x$. After betting on w , the martingale's capital multiplied by a factor of $\frac{d(L(\pi|w)b)F(\pi|w)}{2d(L(\pi|w))F(\pi|w)_b}$. Since $F_1 = \frac{d(L(\pi|w)b)}{d(L(\pi|w))}$ represents the factor by which d 's capital increases while betting on L_π , we only need to show that the remaining factor $F_2 = \frac{F(\pi|w)}{2F(\pi|w)_b}$ doesn't reduce F_1 by much, so that if d succeeds on L_π , D also succeeds on π . We note that $F(\pi|w)$ (the number of length- $3n$ strings whose images have yet to be queried) is at least $2^{3n} - 2^n$, and $F(\pi|w)_b$ (the number of the previously counted strings that begin with bit b) is at most 2^{3n-1} . Therefore, $\frac{F(\pi|w)}{2F(\pi|w)_b} \geq \frac{2^{3n}-2^n}{2 \times 2^{3n-1}} = 1 - 2^{-2n}$. Hence, after betting on all length- $3n$ strings of the form $0^{2|x|}x$, D 's capital is the same as d 's capital after betting on length- n strings multiplied by a factor of at least $(1 - 2^{-2n})^{2^n} \approx e^{-2^{-n}}$. This implies that D 's capital after betting on length- $3n$ strings grows by approximately the same factor by which d 's capital grows after betting on length- n strings. Therefore, if d succeeds on L_π , D succeeds on π . $\square$

Corollary 4.7. *If π is a $\mathbf{p}$ -random permutation, then L_π is a $\mathbf{p}$ -random language.*

We now extend the previous lemma to honest $\mathbf{p}$ -permutation betting games. By Lemma 3.21, honest $\mathbf{p}$ -permutation betting games do not cover PermE and honest $\mathbf{p}_2$ -permutation betting games do not cover PermEXP .

Lemma 4.8. *For any set of permutations $X \subseteq \Pi$, if an honest $\mathbf{p}$ -betting game g succeeds on the set of languages $L_X = \{L_\pi \mid \pi \in X\}$, then there is an honest $\mathbf{p}$ -permutation betting game G that succeeds on X .*

Proof. The proof follows the same strategy as the previous lemma. We use the betting game g to construct G , a permutation betting game that succeeds on $\pi \in X$. The main difference from the previous lemma is in the order that the strings are queried. The order will be directed by simulating the betting game g . Whenever g queries any nonempty string x , G responds by querying the image of $0^{2|x|}x$. The permutation betting game G then bets on the first bit of the image of $0^{2|x|}x$ using g 's bets on the membership of x to determine the proportion placed on the two possible outcomes, just like we did in the previous lemma. Since G will not bet on strings that do not have the form $0^{2|x|}x$, those strings can be queried after g queries all strings of length n . Because g is an honest $\mathbf{p}$ -betting game, it is easy to see that G will be an honest $\mathbf{p}$ -permutation betting game.

The arguments for G 's capital and runtime are almost identical to that of the previous lemmas, the only difference is that we maintain a list of strings that may not be in the standard lexicographic order. $\square$

Corollary 4.9. *If π is an honest $\mathbf{p}$ -betting game random permutation, then L_π is an honest $\mathbf{p}$ -betting game random language.*

Definition 4.10. Given a language L , we define Π_L to be set of permutations

$$\Pi_L = \left\{ \pi \in \Pi \left| \begin{array}{l} \text{for all } n > 0 \text{ and } x \in \{0, 1\}^n, \\ \pi(0^{2n}x) = by \text{ for some } y \in \{0, 1\}^{3n-1} \\ \text{if and only if } L[x] = b \end{array} \right. \right\}.$$

Given a set of languages X , we define Π_X as the set of permutations $\Pi_X = \bigcup_{L \in X} \Pi_L$.

The next lemma is a weak converse to Lemma 4.6.

Lemma 4.11. *For any set of languages $X \subseteq \{0, 1\}^\infty$, if a $\mathbf{p}$ -computable permutation martingale d succeeds on the set of permutations Π_X , then there is a $\mathbf{p}$ -computable martingale that succeeds on X .*

Proof. This proof follows the same general strategy as the proof of Lemma 4.6. So, we will focus on where the proofs differ.

We will use the permutation martingale d to construct a martingale D , that succeeds on the set of languages X . The martingale D operates by simulating d and using d 's bets on strings of the form $0^{2|x|}x$ to bet on the membership of x in the language D is betting on. Let L be the language D is betting on and let π_L be some language in Π_L . We will specify how π_L is constructed when we examine the success set of D .

We now analyze how D bets on all nonempty strings; it does not bet on the empty string. Let $D_L^x(1)$ denote the portion of D 's current capital placed on $x \in \{0,1\}^n$ being a member of L and $D_L^x(0)$ denote the portion placed on x not being a member of L . We set $D_L^x(1)$ to be the portion d places on the first bit of $\pi_L(0^{2n}x)$ being 1 and $D_L^x(0)$ to be the portion d places on the first bit of $\pi_L(0^{2n}x)$ being 0. For simplicity, we use w to denote the string $\pi_L \upharpoonright (0^{2n}x)$. Therefore,

$$D_L^x(b) = \frac{\sum_{s \in \text{free}_b(w)} d(ws)}{|\text{free}(w)|d(w)},$$

where $\text{free}_b(w)$ is the set of strings in $\text{free}(w)$ that begin with bit b . Let $s' \in \text{free}_b(w)$ be a string that minimizes $d(ws')$. Then,

$$D_L^x(b) \geq \frac{|\text{free}_b(w)|}{|\text{free}(w)|} \frac{d(ws')}{d(w)} \geq \frac{(2^{3n-1} - 2^n)}{2^{3n}} \frac{d(ws')}{d(w)} = \frac{1}{2} \left(1 - \frac{1}{2^{2n-1}}\right) \frac{d(ws')}{d(w)}.$$

The first inequality follows by our choice of s' , the second follows because any string of the form $0^{2n}x$ is among the first 2^n string of $\{0,1\}^{3n}$. After betting that x 's characteristic bit is b , the martingale's capital is multiplied by a factor of $2D_L^x(b) \geq \left(1 - \frac{1}{2^{2n-1}}\right) \frac{d(ws')}{d(w)}$. As we have previously shown, the factor $\left(1 - \frac{1}{2^{2n-1}}\right)$ so small that it only contributes a constant factor after betting on all string. So, we only have to focus on the factor $\frac{d(ws')}{d(w)}$. This factor is the factor by which d 's current capital grows after betting on x 's characteristic bit being b .

We now specify how π_L is constructed. When generating the string $w = \pi_L \upharpoonright (0^{2|x|}x)$ which is polynomially longer than $L \upharpoonright x$, we choose images of strings that do not have the form $0^{2|x|}x$ to be any string that minimizes the current capital of the permutation martingale, d . Now we show that D succeeds on any language $L \in X$. It is not hard to see that $\pi_L \in \Pi_L \subseteq \Pi_X$. Therefore, d succeeds on π_L by the hypothesis of this theorem. By the construction of π_L , the only time d 's capital could increase while betting on it, is when it is betting on strings of the form $0^{2|x|}x$. The factor by which d 's current capital is multiplied is $\frac{d(ws')}{d(w)}$, the same factor by which D is multiplied after betting on $L[x]$ being b . Therefore, if d succeeds on π_L , then D also succeeds on L . Since L is an arbitrary language in Π_X , the theorem follows. $\square$

Corollary 4.12. *If L is a p -random language, then Π_L does not have p -permutation measure 0.*

5 Random Permutations for $\text{NP} \cap \text{coNP}$

Bennett and Gill [5] studied the power of random oracles in separating complexity classes. In particular, they showed that $\text{P}^A \neq \text{NP}^A$ relative to a random oracle with probability 1. However, they were not able to separate P from $\text{NP} \cap \text{coNP}$ relative to a random oracle. They also made the observation that if $\text{P}^A = \text{NP}^A \cap \text{coNP}^A$ for a random oracle A , then P^A must include seemingly computationally hard problems such as factorization. They also proved that any non-oracle-dependent language that belongs to P^A with probability 1, also belongs to BPP . As a result,

if $P^A = NP^A \cap coNP^A$ for a random oracle A with probability 1, then these difficult problems in $NP \cap coNP$ would be solvable in BPP. To achieve a separation between P^A and $NP^A \cap coNP^A$, they considered length-preserving permutations on $\{0, 1\}^*$ and showed that $P^\pi \neq NP^\pi \cap coNP^\pi$ for every random permutation π .

Using resource-bounded permutation betting games on the set of all length preserving permutations of $\{0, 1\}^*$, we strengthen the Bennett-Gill permutation separation, proving that $P^\pi \neq NP^\pi \cap coNP^\pi$ for any $\mathbf{p}$ -betting-game random permutations π . More generally, we show that the set of permutations π such that, NP^π is not $DTIME^\pi(2^{kn})$ -bi-immune has $\mathbf{p}$ -permutation-betting-game measure 0. Recall that a language L is bi-immune to a complexity class C if no infinite subset of L or its complement is decidable in C [3, 9].

The following is our main theorem where its first part states that relative to a $\mathbf{p}$ -betting-game random permutation π , there is a language L in $NLIN^\pi \cap coNLIN^\pi$ such that no infinite subset of L or its complement is $DTIME^\pi(2^{kn})$ -decidable.

Theorem 5.1. *1. If π is a $\mathbf{p}$ -betting-game random permutation, then $NLIN^\pi \cap coNLIN^\pi$ contains a $DTIME^\pi(2^{kn})$ -bi-immune language for all $k \geq 1$.*

2. If π is a $\mathbf{p}_2$ -betting-game random permutation, then $NP^\pi \cap coNP^\pi$ contains a $DTIME^\pi(2^{n^k})$ -bi-immune language for all $k \geq 1$.

Our headline result is a corollary of Theorem 5.1.

Corollary 5.2. *If π is a $\mathbf{p}$ -betting-game random permutation, then $P^\pi \neq NP^\pi \cap coNP^\pi$.*

To prove Theorem 5.1, we first define the following test languages. For each $k \geq 1$, define the “half range” test languages

$$\begin{aligned} HRNG_k^\pi &= \{x \mid \exists y \in \{0, 1\}^{k|x|-1}, \pi(0y) = x^k\} \\ &= \{x \mid \forall y \in \{0, 1\}^{k|x|-1}, \pi(1y) \neq x^k\}, \end{aligned}$$

and

$$\begin{aligned} POLYHRNG_k^\pi &= \{x \mid \exists y \in \{0, 1\}^{|x|^k-1}, \pi(0y) = x^{|x|^{k-1}}\} \\ &= \{x \mid \forall y \in \{0, 1\}^{|x|^k-1}, \pi(1y) \neq x^{|x|^{k-1}}\}. \end{aligned}$$

A string $x \in \{0, 1\}^n$ belongs to $HRNG_k^\pi$ if the preimage of x^k (k copies of x) in $\{0, 1\}^{kn}$ begins with 0. If x does not belong to $HRNG_k^\pi$, then the preimage of x^k begins with 1. In either case, the preimage serves as a witness for x . The language $POLYHRNG_k^\pi$ is similar, but we are looking for a preimage in $\{0, 1\}^{n^k}$ of $x^{n^{k-1}}$ (n^{k-1} copies of x). It follows that

$$HRNG_k^\pi \in NLIN^\pi \cap coNLIN^\pi$$

and

$$HRNG_k^\pi \in NTIME^\pi(n^k) \cap coNTIME^\pi(n^k)$$

for all $k \geq 1$.

The following lemma implies Theorem 5.1.

Lemma 5.3. *Let $k \geq 0$.*

1. The set $X = \{\pi \in \Pi \mid \text{HRNG}_{k+3}^\pi \text{ is not } \text{DTIME}(2^{kn})^\pi\text{-immune}\}$ has $O(n)$ -honest $\mathbf{p}$ -permutation-betting-game measure 0.
2. The set $X = \{\pi \in \Pi \mid \text{POLYHRNG}_{k+1}^\pi \text{ is not } \text{DTIME}(2^{n^k})^\pi\text{-immune}\}$ has $O(n^k)$ -honest $\mathbf{p}_2$ -permutation-betting-game measure 0.

Proof. We design a betting game that can succeed on two classes of permutations. The argument breaks down into two main cases.

The betting game succeeds on permutations π that do *not* map strings of the form $1y$ to x^{k+3} . These permutations are relatively straightforward, because one can easily compute the probability that, for all y , $\pi(1y) \neq 0^{(k+3)n}$, i.e. $0^n \notin \text{HRNG}_{k+3}^\pi$. This calculation merely requires examining the status of all length- $(k+3)n$ strings that have been queried. In particular, the order in which the betting game queries strings *does not* affect its ability to wager on this event, making the betting strategy simpler in this case.

We then consider the complementary class of permutations that *do* map some string $1y$ to a string of the form x^{k+3} . Here, y is a *potential witness string*. Unlike the previous case, we exploit the betting game's ability to query strings *out of order*. In particular, we know there exists some permutation TM capable of predicting events of the form $\pi(1y) = x^{k+3}$. Accordingly, our betting game attempts *all* TMs by simulating them on all length- n strings. It is precisely this universal simulation that triggers queries on (potential) witness strings in arbitrary orders. Whenever the simulation would query a witness string, the betting game places a bet on that string *first*.

We show that if, for infinitely many n , a witness string is indeed queried during the simulation, then the betting game wins unbounded capital. On the other hand, if none of the potential witness strings are queried, then there is a TM that makes infinitely many *correct* predictions of the form $\pi(0y) = x^{k+3}$, thereby allowing another betting strategy to succeed.

We now formalize the above outline and construct the required betting game in detail. First, partition X into the two sets

$$X_1 = \{\pi \in X \mid \text{HRNG}_{k+3}^\pi \text{ is finite}\} \quad \text{and} \quad X_2 = \{\pi \in X \mid \text{HRNG}_{k+3}^\pi \text{ is infinite}\}.$$

We now design an honest $\mathbf{p}$ -permutation-betting game, G , that succeeds on all permutations in X_1 as well as in X_2 . To do this, we split G 's initial capital into infinitely many sub-shares according to $a_i = b_i = c_i = \frac{1}{i^2}, i = 1, 2, \dots$. The sub-shares $\{a_i\}$ are used to bet on permutations in X_1 , while the sub-shares $\{b_i, c_i\}$ are reserved for betting on those in X_2 .

Succeeding on X_1 . For a randomly selected permutation π , consider the event $\mathcal{A}_n$, that $0^n \notin \text{HRNG}_{k+3}^\pi$ i.e., $\pi(1y) = 0^{n(k+3)}$, for some string y . The probability of $\mathcal{A}_n$ is $1/2$ and it only depends on images of length- $(k+3)n$ strings. To keep track of which length- $(k+3)n$ strings have been queried, we let $\omega \in (\{\{0, 1\}^{(k+3)n}\} \cup \{\star\})^{2^{(k+3)n}}$ encode the current *status* of each strings. Specifically, if the i th length- $(k+3)n$ string has already been queried and mapped to some string z , then we set $\omega[i] = z \in \{0, 1\}^{(k+3)n}$; otherwise, if it has not been queried yet, we set $\omega[i] = \star$.

Recall that G is our overall betting game, which updates its capital shares a_i through a “sub-game” G_{a_i} whenever $i \geq n$. We define

$$G_{a_i}(\omega) = \frac{a_i}{\Pr(\mathcal{A}_n)} \Pr(\mathcal{A}_n \mid \omega).$$

Let $\omega^{i \rightarrow b}$ denote the string ω with its i th component replaced by $b \in \{\{0, 1\}^{(k+3)n}\} \cup \{\star\}$. We define $\text{free}(\omega)$ to be the set of all length- $(k+3)n$ strings available as an image. It is routine to verify that:

$$G_{a_i}(\omega^{i \rightarrow \star}) = \frac{1}{|\text{free}(\omega^{i \rightarrow \star})|} \sum_{b \in \text{free}(\omega^{i \rightarrow \star})} G_{a_i}(\omega^{i \rightarrow b}).$$

Hence, the summation of all G_{a_i} forms a betting game.

Since $\Pr(\mathcal{A}_n) = \frac{1}{2}$, it is easy to verify that whenever ω represents a configuration in which $\mathcal{A}_n$ is true, we have

$$G_{a_i}(\omega) = 2a_i.$$

Thus every time $\mathcal{A}_n$ occurs, the subgame G_{a_i} doubles its capital from a_i to $2a_i$. For every permutation $\pi \in X_1$, $\mathcal{A}_n$ holds for all but finitely many n , so infinitely many a_i 's grow unboundedly. Therefore, G succeeds on all $\pi \in X_1$.

The “subgame” G_{a_i} can be computed in $O(2^{(k+3)n})$ time by examining the mapping of every string of length $(k+3)n$. Moreover, the order in which G queries length- $2n$ strings has no impact on its success for $\pi \in X_1$. As we shall see next, this changes when dealing with permutations in X_2 .

Succeeding on X_2 . Given $\pi \in X_2$, we know HRNG_{k+3}^π is not $\text{DTIME}(2^{kn})^\pi$ -immune. Thus there is some $\text{DTIME}(2^{kn})^\pi$ oracle TM that recognizes an infinite subset of HRNG_{k+3}^π . A string $y \in \{0, 1\}^{k(n+1)-1}$ is a *witness* for the membership of a length- n string x in HRNG_{k+3}^π if $\pi(0y) = x^{(k+1)}$. We simulate the first n such $\text{DTIME}(2^{kn})^\pi$ machines on all length- n strings. Two cases arise:

(1) We query witnesses of some $x \in \text{HRNG}_{k+3}^\pi$ for infinitely many n during the simulation phase. Let $Q_n \subseteq \{0, 1\}^{(k+3)n-1}$ be the set of strings that might witness membership of any length- n string x queried in the simulation. We use a portion of b_n to bet on the event that $y \in Q_n$ is indeed a witness to $x \in \text{HRNG}_{k+3}^\pi$. For large n , $|Q_n| < n^2 2^{(k+1)n}$ because each of the n rounds simulates at most n distinct $\text{DTIME}(2^{kn})^\pi$ machines on all length- n inputs. We split b_n evenly among the $2^n \times |Q_n|$ possible string-witness pairs; a winning bet multiplies that share by at least $2^{(k+3)n} - n^2 2^{(k+2)n}$ (because the TMs running so far could have queried at most $n^2 2^{(k+2)n}$ length- $(k+3)n$ strings). Hence the capital used per event jumps to $(\frac{b_n}{n^2 2^{(k+2)n}})(2^{(k+3)n} - n^2 2^{(k+2)n}) = \omega(2^{0.9n})$, and since one of these events arises infinitely often, the betting game succeeds.

(2) Witness strings are queried only finitely many times. Eventually, none of the TMs ever queries a witness string. At that stage, after simulating the TMs on length- n strings, the betting game employs c_i to wager on any $\text{DTIME}(2^{kn})^\pi$ TM M_i that has not made a mistake and accepts some $x \in \{0, 1\}^n$. Let B_x be the event $x \in \text{HRNG}_{k+3}^\pi$, i.e. there exists y with $\pi(0y) = x^{(k+1)}$. For a random permutation, $\Pr(B_x) = \frac{1}{2}$. After simulating all TMs on length- n strings *without* querying any witness for x , the conditional probability of B_x remains at least $\frac{2^{(k+3)n} - n^2 2^{(k+2)n}}{2^{(k+3)n}} \approx \frac{1}{2}$. Hence the betting game approximately doubles c_i each time $M_i(x)$ is accepted. Because we are guaranteed that some M_i correctly decides an infinite subset of HRNG_{k+3}^π , this betting also succeeds.

Clearly, this construction is implementable in $\text{DTIME}(2^{O(n)})$. Notice moreover that while querying strings of length n , the betting game places wagers only on strings of length $(k+3)n$, making it an $(k+3)n$ -honest p-permutation betting game.

Sketch for part (2). The second statement uses $\text{POLYHRNG}_{k+1}^\pi$ in place of HRNG_{k+3}^π , and the class $\text{DTIME}(2^{n^k})^\pi$ replaces $\text{DTIME}(2^{kn})^\pi$. One repeats the same betting-game strategy but scales the length parameters and the honesty parameter accordingly; the resulting game is $O(n^{k+1})$ -honest in $\text{DTIME}(2^{n^{O(1)}})$. All other details are essentially unchanged. Therefore, the set of permutations for which $\text{POLYHRNG}_{k+1}^\pi$ is not $\text{DTIME}(2^{n^k})^\pi$ -immune has $\mathbf{p}_2$ -permutation-betting-game measure 0. $\square$

By symmetry of $\text{NLIN}^\pi \cap \text{coNLIN}^\pi$ and $\text{NTIME}^\pi(n^k) \cap \text{coNTIME}^\pi(n^k)$, Lemma 5.3 also applies to the complement of HRNG_{k+3}^π and $\text{POLYHRNG}_{k+1}^\pi$. Therefore, both languages are bi-immune and Theorem 5.1 follows.

Combining Lemma 5.3 with Proposition 3.23 also gives the following corollary. In the next section we will prove more results about $\mathbf{p}$ space-random permutations.

Corollary 5.4. 1. If π is a pspace -random permutation, then $\text{NLIN}^\pi \cap \text{coNLIN}^\pi$ contains a $\text{DTIME}^\pi(2^{kn})$ -bi-immune language for all $k \geq 1$.

2. If π is a p_2space -random permutation, then $\text{NP}^\pi \cap \text{coNP}^\pi$ contains a $\text{DTIME}^\pi(2^{n^k})$ -bi-immune language for all $k \geq 1$.

6 Random Permutations for $\text{NP} \cap \text{coNP}$ versus Quantum Computation

Bennett, Bernstein, Brassard, and Vazirani [4] showed that $\text{NP}^\pi \cap \text{coNP}^\pi \not\subseteq \text{BQTIME}^\pi(o(2^{n/3}))$ relative to a random permutation π with probability 1. In this section we investigate how much of their result holds relative to individual random oracles at the space-bounded level.

We begin with a general lemma about test languages and QTMs. We write $\text{PP}\Pi_{\leq n} = \{g \in \text{PP}\Pi \mid |g| \leq 2^{n+1} - 1\}$ for all prefix partial permutations defined on strings in $\{0, 1\}^{\leq n}$. For a string s_i in the standard enumeration, we write $g \upharpoonright s_i$ for the length i prefix of g . In other words, $g \upharpoonright s_i = [g(s_0), \dots, g(s_{i-1})]$.

Lemma 6.1. Let π be a permutation with an associated test language L_π and let $p(n)$ be a linear function (polynomial function, respectively). If for some oracle QTM M the following conditions hold, then π is not a pspace -random (p_2space -random, respectively) permutation.

1. The membership of 0^n in L_π depends on the membership of the strings of length at most $p(n)$.
2. M^π decides L_π with error probability δ , for some constant $0 < \delta < 1$, and queries only strings of length at most $p(n)$.
3. For any partial prefix permutation $\rho \in \text{PP}\Pi_{\leq p(n)}$, the conditional probability

$$\Pr_{|\psi|=p(n)} \left[M^\psi(0^n) = L_\psi(0^n) \mid \rho \sqsubseteq \psi \right]$$

is computable in $O(2^{O(n)})$ space ($O(2^{n^{O(1)}})$ space, respectively).

4. For some constant $1 > \epsilon > \delta$ and for all but finitely many n ,

$$\Pr_{|\psi|=l(n)} \left[M^\psi(0^n) = L_\psi(0^n) \mid \pi \upharpoonright 0^n \sqsubseteq \psi \right] < 1 - \epsilon.$$

Proof. Let $n_0 < n_1 < n_2 < \dots$ be a sequence of numbers such that $p(n_i) < n_{i+1}$ for all $i \geq 0$. Based on the statement of the lemma, $M^\psi(0^{n_{j-1}})$ cannot query any string of length n_j and the membership of $0^{n_{j-1}}$ does not depend on the membership of any string of length greater than n_j . Let n_k be the first number in the sequence above such that for any $n > n_k$ we have $\Pr_{|\psi|=l(n)} [M^\psi(0^n) = L_\psi(0^n) \mid \pi \upharpoonright 0^n \sqsubseteq \psi] < 1 - \epsilon$ and define the following martingale:

$$d(\rho) = \begin{cases} 1 & \text{if } |\rho| \leq 2^{n_k+1} - 1 \\ \frac{d(\rho \upharpoonright 0^{n_{j-1}})}{\Pr(\psi \upharpoonright \rho \upharpoonright 0^{n_{j-1}})} \Pr(\psi \mid \rho) & \text{if } 2^{n_{j-1}} < |\rho| \leq 2^{n_j+1} - 1, \text{ for some } j > k \end{cases}$$

where $\Pr(\psi \mid \nu)$ is the probability that $M^\psi(0^n) = L_\psi(0^n)$ given that $\nu \sqsubseteq \psi$. The third condition in the statement of the lemma implies that this martingale is pspace computable (p_2space computable,

respectively). Now consider a permutation π that satisfies the conditions of the lemma. Then for $j > k$ we have:

$$d(\pi \upharpoonright 0^{n_j}) = \frac{d(\pi \upharpoonright 0^{n_{j-1}})}{\Pr(\psi \mid \pi \upharpoonright 0^{n_{j-1}})} \Pr(\psi \mid \pi \upharpoonright 0^{n_j}) \geq \frac{d(\pi \upharpoonright 0^{n_{j-1}})}{1 - \epsilon} (1 - \delta)$$

The last inequality holds because it follows from the first two conditions of the lemma that $\Pr(\psi \mid \pi \upharpoonright 0^{n_j}) \geq 1 - \delta$. By repeating this process, we can see that d succeeds on π . $\square$

In the following theorem, we use Lemma 6.1 to extend the result by Bennett, Bernstein, Brassard, and Vazirani [4] to $\mathbf{p}_2\mathbf{space}$ -random permutations.

Theorem 6.2. *If π is a $\mathbf{p}_2\mathbf{space}$ -random permutation, then $\mathbf{NLIN}^\pi \cap \mathbf{coNLIN}^\pi$ is not contained in $\mathbf{BQP}^\pi$.*

Proof. Let M be a $\mathbf{BQP}$ oracle machine running in time $t(n)$. From Bennett et al. [4] it follows that that relative to a random permutation π , M fails to decide whether 0^n is in $\mathbf{HRNG}_n^\pi$ with probability at least $1/8$. We can compute

$$\Pr_{|\psi|=t(n)} \left[M^\psi(0^n) = \mathbf{HRNG}_n^\psi(0^n) \mid \rho \sqsubseteq \psi \right]$$

in space $2^{n^{O(1)}}$. The Theorem follows from Lemma 6.1. $\square$

We now refine the previous result by considering more restricted quantum machines that only query strings of $O(n)$ length. This restriction allows us to extend the result to machines with running time $o(2^{n/3})$, analogous to the result of Bennett et al. [4]. Whether this extension holds without the restriction on query length remains an open problem.

Theorem 6.3. *If π is a $\mathbf{pspace}$ -random permutation and $T(n) = o(2^{n/3})$, then $\mathbf{NLIN}^\pi \cap \mathbf{coNLIN}^\pi$ is not contained $\mathbf{BQTIME}^{\pi, O(n)\text{-honest}}(T(n))$.*

Proof. Under the honesty condition, we can compute the necessary conditional probability in $2^{O(n)}$ space and apply Lemma 6.1. $\square$

Together, these theorems extend the classical separation of Bennett et al. [4] to individual space-bounded random permutations, both in the general and the honest-query setting.

7 Random Oracles for $\mathbf{NP} \cap \mathbf{coNP}$ and 0-1 Laws for Measure in $\mathbf{EXP}$

Tardos [29] used the characterizations

$$\mathbf{BPP} = \mathbf{ALMOST-P} = \{A \mid \Pr_R[A \in \mathbf{P}^R] = 1\} \quad [5]$$

and

$$\mathbf{AM} = \mathbf{ALMOST-NP} = \{A \mid \Pr_R[A \in \mathbf{NP}^R] = 1\} \quad [26]$$

to prove the following conditional theorem separating $\mathbf{P}$ from $\mathbf{NP} \cap \mathbf{coNP}$ relative to a random oracle.

Theorem 7.1. (Tardos [29]) *If $\mathbf{AM} \cap \mathbf{coAM} \neq \mathbf{BPP}$, then $\mathbf{P}^R \neq \mathbf{NP}^R \cap \mathbf{coNP}^R$ for a random oracle R with probability 1.*

Recently, Ghosal et al. [10] used non-interactive zero-knowledge (NIZK) proofs to prove a similar conditional theorem.

Theorem 7.2. (Ghosal et al. [10]) *If $UP \not\subseteq RP$, then $P^R \neq NP^R \cap coNP^R$ for a random oracle R with probability 1.*

In this section we use Theorems 7.1 and 7.2 to connect the open problem of P versus $NP \cap coNP$ relative to a random oracle to open questions about the resource-bounded measure of complexity classes within EXP . In particular, we relate the problem to measure 0-1 laws and measurability in EXP . First, we need the following derandomization lemma. The first two parts follow from previous work, while the third part of the lemma is a new observation as far as we know, though its proof uses the techniques from the proofs of the first two parts.

Lemma 7.3. 1. *If $\mu_p(NP) \neq 0$, then $BPP \subseteq NP \cap coNP = AM \cap coAM$.*

2. *If $\mu_p(UP \cap coUP) \neq 0$, then $BPP \subseteq UP \cap coUP$.*

3. *If $\mu_p(FewP) \neq 0$, then $BPP \subseteq FewP \cap coFewP$.*

Proof. 1. $\mu_p(NP) \neq 0$ implies $NP = AM$ [17]. Then $BPP \subseteq AM \cap coAM = NP \cap coNP$.

2. If $\mu_p(UP \cap coUP) \neq 0$, then the UP -machine hypothesis holds, which implies $BPP \subseteq UP \cap coUP$ [14].

3. If $\mu_p(FewP) \neq 0$, then there is a p -random language R in $FewP$ [2]. All but finitely many witnesses for membership in R have high circuit complexity [14, 17]. By the longest runs theorem [12], there is guaranteed to be a string in R within the first $2n$ strings of $\{0, 1\}^n$, for all but finitely many n . We guess one of these strings and a witness. If we find a valid witness, we use it to build a pseudorandom generator [18, 26] and derandomize BPP [14]. There are at most a polynomial number of witnesses, placing $BPP \subseteq FewP \cap coFewP$. $\square$

In the following theorem, we have three hypotheses where a complexity class X is assumed to be not equal to EXP and the p -measure of a subclass of X is concluded to be 0.

Theorem 7.4. *Suppose that $P^R = NP^R \cap coNP^R$ for a random oracle R with probability 1. Then all of the following hold:*

1. $NP \neq EXP \Rightarrow \mu_p(NP \cap coNP) = 0$.

2. $UP \neq EXP \Rightarrow \mu_p(UP \cap coUP) = 0$.

3. $FewP \neq EXP \Rightarrow \mu_p(UP) = 0$.

Proof. 1. Suppose $\mu_p(NP \cap coNP) \neq 0$ and $NP \neq EXP$. From Lemma 7.3 we have $BPP \subseteq NP \cap coNP = AM \cap coAM$. Therefore $BPP \subseteq NP \neq EXP$, so $\mu_p(BPP) = 0$ by the zero-one law for BPP [30]. Since $AM \cap coAM$ and BPP have different p -measures, the classes are not equal. The result follows from Theorem 7.1.

2. Suppose $\mu_p(UP \cap coUP) \neq 0$ and $UP \neq EXP$. From Lemma 7.3 we have $BPP \subseteq UP \cap coUP$. Therefore $BPP \subseteq UP \neq EXP$, so $\mu_p(BPP) = 0$ by the zero-one law. Therefore $UP \cap coUP \not\subseteq BPP$, so $UP \not\subseteq RP$. The result follows from Theorem 7.2.

3. Suppose $\mu_p(\text{UP}) \neq 0$ and $\text{FewP} \neq \text{EXP}$. Since $\text{UP} \subseteq \text{FewP}$, we have $\mu_p(\text{FewP}) \neq 0$ and Lemma 7.3 implies $\text{BPP} \subseteq \text{FewP} \cap \text{coFewP}$. Therefore $\text{BPP} \subseteq \text{FewP} \neq \text{EXP}$, so $\mu_p(\text{BPP}) = 0$ by the zero-one law. Therefore $\text{UP} \not\subseteq \text{BPP}$, so $\text{UP} \not\subseteq \text{RP}$. The result follows from Theorem 7.2. $\square$

Theorem 7.4 has the following corollary about measure 0-1 laws in EXP . We recall the definitions $\mu(X \mid \text{EXP}) = 0$ if $\mu_{p_2}(X \cap \text{EXP}) = 0$ and $\mu(X \mid \text{EXP}) = 1$ if $\mu_{p_2}(X^c \mid \text{EXP}) = 0$ [20].

Corollary 7.5. *Suppose that $\text{P}^R = \text{NP}^R \cap \text{coNP}^R$ for a random oracle R with probability 1. Then all of the following hold:*

1. $\mu(\text{NP} \cap \text{coNP} \mid \text{EXP}) \in \{0, 1\}$.
2. $\mu(\text{UP} \cap \text{coUP} \mid \text{EXP}) \in \{0, 1\}$.
3. $\mu(\text{UP} \mid \text{EXP}) = 0$ or $\mu(\text{FewP} \mid \text{EXP}) = 1$.

Proof. The two key facts we need in this proof are that for any class $X \subseteq \text{EXP}$:

- (a) If X is closed under finite union and intersection, then $X = \text{EXP}$ if and only if $\mu(X \mid \text{EXP}) = 1$ [21, 27].
- (b) If $\mu_p(X) = 0$, then $\mu(X \mid \text{EXP}) = 0$ [20].

We write each implication in Theorem 7.4 as a disjunction and apply the above facts:

1. We have $\text{NP} = \text{EXP} \Leftrightarrow \text{NP} \cap \text{coNP} = \text{EXP} \Leftrightarrow \mu(\text{NP} \cap \text{coNP} \mid \text{EXP}) = 1$ or $\mu_p(\text{NP} \cap \text{coNP}) = 0 \Rightarrow \mu(\text{NP} \cap \text{coNP} \mid \text{EXP}) = 0$.
2. We have $\text{UP} = \text{EXP} \Leftrightarrow \text{UP} \cap \text{coUP} = \text{EXP} \Leftrightarrow \mu(\text{UP} \cap \text{coUP} \mid \text{EXP}) = 1$ or $\mu_p(\text{UP} \cap \text{coUP}) = 0 \Rightarrow \mu(\text{UP} \cap \text{coUP} \mid \text{EXP}) = 0$.
3. We have $\text{FewP} = \text{EXP} \Leftrightarrow \mu(\text{FewP} \mid \text{EXP}) = 1$ or $\mu_p(\text{UP}) = 0 \Rightarrow \mu(\text{UP} \mid \text{EXP}) = 0$.

$\square$

In the third case of Corollary 7.5, we almost have a 0-1 law for UP . Can a full 0-1 law be obtained?

The contrapositives of the implications in Corollary 7.5 show that the random oracle question for P versus $\text{NP} \cap \text{coNP}$ is resolved under nonmeasurability hypotheses. A complexity class X is defined to be *not measurable* in EXP if $\mu(X \mid \text{EXP}) \neq 0$ and $\mu(X \mid \text{EXP}) \neq 1$ [21, 27].

Corollary 7.6. *1. If $\text{NP} \cap \text{coNP}$ is not measurable in EXP , then $\text{P}^R \neq \text{NP}^R \cap \text{coNP}^R$ for a random oracle R with probability 1.*

2. If $\text{UP} \cap \text{coUP}$ is not measurable in EXP , then $\text{P}^R \neq \text{NP}^R \cap \text{coNP}^R$ for a random oracle R with probability 1.

3. If UP and FewP are both not measurable in EXP , then $\text{P}^R \neq \text{NP}^R \cap \text{coNP}^R$ for a random oracle R with probability 1.

On the other hand, if the consequence of Corollary 7.6 can be proved with measure in EXP , then we would have $\text{BPP} \neq \text{EXP}$, which implies $\mu(\text{BPP} \mid \text{EXP}) = 0$ by the 0-1 law for BPP [30].

Theorem 7.7. *If $\{A \mid \text{P}^A = \text{NP}^A \cap \text{coNP}^A\}$ has measure 0 in EXP , then $\mu(\text{BPP} \mid \text{EXP}) = 0$.*

Proof. Let A be $\leq_T^P$ -complete for EXP. Then

$$\text{NP}^A \cap \text{coNP}^A \subseteq \text{EXP} \subseteq \text{P}^A \subseteq \text{NP}^A \cap \text{coNP}^A.$$

Therefore the $\leq_T^P$ -complete sets are a subset of $\{A \mid \text{P}^A = \text{NP}^A \cap \text{coNP}^A\}$, so they have measure 0 in EXP, which implies $\text{BPP} \neq \text{EXP}$ [7] and $\mu(\text{BPP} \mid \text{EXP}) = 0$. $\square$

These results suggest that resolving whether $\text{P}^R = \text{NP}^R \cap \text{coNP}^R$ relative to a random oracle R requires a deeper understanding of the resource-bounded measurability within EXP of fundamental subclasses such as BPP, NP, UP, and FewP.

8 Conclusion

We have introduced resource-bounded random permutations and shown that $\text{P}^\pi \neq \text{NP}^\pi \cap \text{coNP}^\pi$ for all p -betting-game random permutations. We remark that all of the results in Sections 5 and 6 about $\text{NLIN} \cap \text{coNLIN}$ and $\text{NP} \cap \text{coNP}$ hold for their unambiguous versions $\text{ULIN} \cap \text{coULIN}$ and $\text{UP} \cap \text{coUP}$, respectively. An interesting open problem is whether our main theorem can be improved from betting-game random permutations to random permutations.

Question 8.1. *Does $\text{P}^\pi \neq \text{NP}^\pi \cap \text{coNP}^\pi$ for a p -random permutation π ?*

More generally, the relative power of permutation martingales versus betting games should be investigated.

Question 8.2. *Are polynomial-time permutation martingales and permutation betting games equivalent?*

We proved two restricted versions of the Bennett et al. [4] random permutation separation. Does the full version hold relative to individual random permutations?

Question 8.3. *If π is a pspace -random permutation and $T(n) = o(2^{n/3})$, is $\text{NLIN}^\pi \cap \text{coNLIN}^\pi$ not contained in $\text{BQTIME}^\pi(T(n))$?*

References

- [1] K. Ambos-Spies and E. Mayordomo. Resource-bounded measure and randomness. In A. Sorbi, editor, *Complexity, Logic and Recursion Theory*, Lecture Notes in Pure and Applied Mathematics, pages 1–47. Marcel Dekker, New York, N.Y., 1997. doi:10.1201/9780429187490-1.
- [2] 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. doi:10.1016/s0304-3975(95)00260-x.
- [3] J. L. Balcázar and U. Schöning. Bi-immune sets for complexity classes. *Mathematical Systems Theory*, 18:1–10, 1985. doi:10.1007/bf01699457.
- [4] C. H. Bennett, E. Bernstein, G. Brassard, and U. V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. arXiv:quant-ph/9701001, doi:10.1137/s0097539796300933.
- [5] C. H. Bennett and J. Gill. Relative to a random oracle A , $\text{P}^A \neq \text{NP}^A \neq \text{co-NP}^A$ with probability 1. *SIAM Journal on Computing*, 10:96–113, 1981. doi:10.1137/0210008.

- [6] R. V. Book, J. H. Lutz, and K. W. Wagner. An observation on probability versus randomness with applications to complexity classes. *Mathematical Systems Theory*, 27:201–209, 1994. doi:10.1007/bf01578842.
- [7] H. Buhrman, D. van Melkebeek, K. W. Regan, D. Sivakumar, and M. Strauss. A generalization of resource-bounded measure, with application to the BPP vs. EXP problem. *SIAM Journal on Computing*, 30(2):576–601, 2001. doi:10.1137/S0097539798343891.
- [8] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976. doi:10.1109/TIT.1976.1055638.
- [9] P. Flajolet and J. Steyaert. On sets having only hard subsets. In *Proc. 2nd Colloq. on Automata, Languages, and Programming, Lecture Notes in Computer Science*, volume 14, pages 446–457. Springer-Verlag, Berlin, 1974. doi:10.1007/978-3-662-21545-6_34.
- [10] Riddhi Ghosal, Yuval Ishai, Alexis Korb, Eyal Kushilevitz, Paul Lou, and Amit Sahai. Hard languages in $\text{NP} \cap \text{coNP}$ and NIZK proofs from unstructured hardness. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 1243–1256, New York, NY, USA, 2023. Association for Computing Machinery. doi:10.1145/3564246.3585119.
- [11] R. C. Harkins and J. M. Hitchcock. Exact learning algorithms, betting games, and circuit lower bounds. *ACM Transactions on Computation Theory*, 5(4):article 18, 2013. doi:10.1145/2539126.2539130.
- [12] R. C. Harkins, J. M. Hitchcock, and A. Pavan. Stronger reductions and isomorphism of complete sets. *Computability*, 3(2):91–104, 2014. doi:10.3233/COM-140028.
- [13] J. M. Hitchcock and J. H. Lutz. Why computational complexity requires stricter martingales. *Theory of Computing Systems*, 39(2):277–296, 2006. doi:10.1007/s00224-005-1135-4.
- [14] J. M. Hitchcock and A. Pavan. Hardness hypotheses, derandomization, and circuit complexity. *Computational Complexity*, 17(1):119–146, 2008. doi:10.1007/s00037-008-0241-5.
- [15] J. M. Hitchcock, A. Sekoni, and H. Shafei. Polynomial-time random oracles and separating complexity classes. *ACM Transactions on Computation Theory*, 13(1), 2021. doi:10.1145/3434389.
- [16] J. M. Hitchcock, A. Sekoni, and H. Shafei. Random permutations in computational complexity. In *Proceedings of the 50th International Symposium on Mathematical Foundations of Computer Science (MFCS 2025)*, volume 345 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 58:1–58:17, 2025. doi:10.4230/LIPIcs.MFCS.2025.58.
- [17] R. Impagliazzo and P. Moser. A zero-one law for RP and derandomization of AM if NP is not small. *Information and Computation*, 207(7):787–792, 2009. doi:10.1016/j.ic.2009.02.002.
- [18] R. Impagliazzo and A. Wigderson. Randomness vs. time: Derandomization under a uniform assumption. *Journal of Computer and System Sciences*, 63:672–688, 2001. doi:10.1006/jcss.2001.1780.
- [19] A. Klivans and D. van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM Journal on Computing*, 31(5):1501–1526, 2002. doi:10.1137/s0097539700389652.

- [20] J. H. Lutz. Almost everywhere high nonuniform complexity. *Journal of Computer and System Sciences*, 44(2):220–258, 1992. doi:10.1016/0022-0000(92)90020-j.
- [21] J. H. Lutz. The quantitative structure of exponential time. In L. A. Hemaspaandra and A. L. Selman, editors, *Complexity Theory Retrospective II*, pages 225–254. Springer-Verlag, 1997. doi:10.1007/978-1-4612-1872-2_10.
- [22] J. H. Lutz and W. J. Schmidt. Circuit size relative to pseudorandom oracles. *Theoretical Computer Science*, 107(1):95–120, 3 1993. doi:10.1016/0304-3975(93)90256-s.
- [23] P. Martin-Löf. The definition of random sequences. *Information and Control*, 9:602–619, 1966. doi:10.1016/s0019-9958(66)80018-9.
- [24] W. Merkle, J. S. Miller, A. Nies, J. Reimann, and F. Stephan. Kolmogorov-Loveland randomness and stochasticity. *Annals of Pure and Applied Logic*, 138(1–3):183–210, 2006. doi:10.1016/j.apal.2005.06.011.
- [25] A. A. Muchnik, A. L. Semenov, and V. A. Uspensky. Mathematical metaphysics of randomness. *Theoretical Computer Science*, 207(2):263 – 317, 1998. doi:10.1016/S0304-3975(98)00069-3.
- [26] N. Nisan and A. Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994. doi:10.1016/s0022-0000(05)80043-1.
- [27] K. W. Regan, D. Sivakumar, and J. Cai. Pseudorandom generators, measure theory, and natural proofs. In *Proceedings of the 36th Symposium on Foundations of Computer Science*, pages 26–35. IEEE Computer Society, 1995. doi:10.1109/SFCS.1995.492459.
- [28] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, February 1978. doi:10.1145/359340.359342.
- [29] G. Tardos. Query complexity, or why is it difficult to separate $NP^A \cap coNP^A$ from P^A by random oracles A ? *Combinatorica*, 9(4):385–392, 1989. doi:10.1007/BF02125350.
- [30] D. van Melkebeek. The zero-one law holds for BPP. *Theoretical Computer Science*, 244(1–2):283–288, 2000. doi:10.1016/s0304-3975(00)00191-2.
- [31] J. Ville. *Étude Critique de la Notion de Collectif*. Gauthier-Villars, Paris, 1939.