One-Way Functions and Polynomial-Time Dimension

Satyadev Nandakumar
Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, India

Subin Pulari
National Research University Higher School of Economics Moscow, Russia

Akhil S
Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, India

Suronjona Sarma
Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, India

Abstract

The theory of randomness in computation has developed around several viewpoints on randomness and information, including statistical tests, compression schemes, betting strategies, and efficiently samplable sources. At the level of computability, several of these viewpoints turn out to be equivalent. A fundamental question is how robust these viewpoints remain when the underlying algorithms are subject to computational resource bounds. This paper studies this question for two polynomial-time notions of information density for infinite binary sequences. Polynomial-time dimension, denoted $\dim_{\mathrm{P}}$, quantifies information density using polynomial-time betting strategies called s-gales. Polynomial-time Kolmogorov complexity rate, denoted $\mathcal{K}_{\mathrm{poly}}$, gives a compression-based notion of polynomial-time information density using polynomial-time descriptions. Hitchcock and Vinodchandran (CCC 2004) showed that $\dim_{\mathrm{P}}(X) \geq \mathcal{K}_{\mathrm{poly}}(X)$ for every sequence X, and asked whether equality always holds. This question was later also posed by Stull.

Our main result proves a duality between the non-robustness of these polynomial-time information-density notions and the existence of one-way functions. Assuming one-way functions exist, we construct a polynomial-time samplable distribution over infinite sequences such that, with probability 1, the sampled sequence X satisfies $\dim_{\mathrm{P}}(X) > \mathcal{K}_{\mathrm{poly}}(X)$ by a uniform gap. Conversely, we show that if some polynomial-time samplable distribution yields such an almost-sure uniform separation, then infinitely-often one-way functions exist. Thus, separations between these two polynomial-time information-density notions over efficiently samplable sources lie at the same frontier as one-way functions, with the reverse direction yielding infinitely-often one-way functions. This duality gives a negative answer, assuming one-way functions, to the open question posed by Hitchcock, Vinodchandran, and Stull. Furthermore, we show that there are individual sequences witnessing separations between $\dim_{\mathrm{P}}$ and $\mathcal{K}_{\mathrm{poly}}$, and that the gap can be made arbitrarily close to 1. We also establish analogous bounds for strong polynomial-time dimension and asymptotic upper polynomial-time Kolmogorov complexity rates. The main technical challenge is to connect finite-length pseudorandomness assumptions with asymptotic information-density measures on infinite sequences. Our proof addresses this by developing several new constructions and arguments involving probabilistic tools such as the Borel–Cantelli lemma, Kolmogorov's inequality for martingales, and techniques for approximating probabilities of efficiently samplable distributions. The work shows that the question of non-robustness for polynomial-time information-density notions, which is prima facie different from standard questions about randomness, pseudorandomness, and cryptography, is in fact intimately related to the existence of one-way functions.

2012 ACM Subject Classification Theory of computation → Pseudorandomness and derandomization; Theory of computation → Complexity classes; Theory of computation → Cryptographic primitives

Keywords and phrases Polynomial-time dimension, One-way functions, Resource bounded randomness, Kolmogorov complexity, Polynomial-time martingales

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.44

Category RANDOM

Related Version *ECCC Report*: https://eccc.weizmann.ac.il/report/2025/028/

Funding *Subin Pulari*: Supported by the Basic Research Program at HSE University and this paper was prepared in the framework of this program.
Suronjona Sarma: Supported by the Visvesvaraya PhD Scheme Scholarship. This research work was partially supported by the Research-I Foundation of the Department of Computer Science and Engineering, IIT Kanpur for conference travel.

Acknowledgements The authors would like to thank CS Bhargav, Laurent Bienvenu, John Hitchcock, Elvira Mayordomo, Jaikumar Radhakrishnan, Mahesh Sreekumar Rajasree, Bruno Bauwens and Marius Zimand for helpful comments and discussions.

1 Introduction

An infinite binary sequence can contain regularity that is exposed in different ways. A statistical test may detect a deviation from randomness, a short description may compress many of its prefixes, or a betting strategy may turn predictable patterns into unbounded capital. These perspectives lead both to qualitative notions of randomness and quantitative measures of how much randomness remains visible to a given computational model.

This paper studies one such measure: polynomial-time dimension. We address the longstanding open question of whether its betting and compression formulations continue to agree. We give a negative answer under the assumption that one-way functions exist. The connection is not limited to deriving separations from cryptographic assumptions: we also prove a converse showing that such separations over efficiently samplable sources imply infinitely-often one-way functions.

Constructive dimension is an algorithmic analogue of classical Hausdorff dimension for individual infinite sequences. It was introduced by Lutz [27] via *unfair constructive betting strategies* on infinite sequences, called s-gales. Intuitively, an s-gale reads a sequence bit by bit and repeatedly bets its current capital on the next bit. The parameter s controls the payoff rule of the game. When s is smaller, the game is less favorable to the gambler, so success requires the gambler to exploit stronger regularities in the sequence. The gale succeeds on a sequence if its capital becomes arbitrarily large along that sequence. The constructive dimension of a sequence is the infimum over all values of s for which some constructive s-gale succeeds. Thus, constructive dimension is a threshold measure of how difficult it is for constructive betting strategies to extract profit from the sequence.

Lutz [28] and Mayordomo [29] gave an equivalent characterization of constructive dimension of a sequence in terms of the Kolmogorov complexity of its prefixes. Here $K(x)$ denotes the length of the shortest program that outputs the finite string x. In this formulation, the information density of a sequence is measured by comparing the shortest description lengths of its prefixes with their corresponding lengths. More precisely, this compression view is governed by the lower asymptotic rate of prefix complexity, namely the liminf of $K(X \upharpoonright n)/n$, where $X \upharpoonright n$ denotes the first n bits of X. Prefixes that admit much shorter descriptions indicate lower information density, while prefixes that resist compression indicate higher information density. Thus, for constructive dimension, the betting formulation and the compression formulation define the same notion of information density. These equivalent characterizations establish the robustness of constructive dimension as a mathematical concept. They also provide multiple viewpoints that are useful in different contexts for studying randomness and information density. This leaves a natural robustness problem in the polynomial-time setting: whether the compression characterization continues to match the betting characterization once both are restricted to efficient computation.

Polynomial-time dimension is the analogue of constructive dimension at the level of feasible or polynomial-time computation. As above, there are multiple approaches towards quantifying polynomial-time density of information. The *compressibility* approach uses the time-bounded Kolmogorov complexity of finite prefixes of infinite sequences. The resulting quantity, denoted $\mathcal{K}_{\text{poly}}$, is obtained from the lower asymptotic rate of prefix complexity when the programs producing the prefixes are required to run in polynomial time. In other words, $\mathcal{K}_{\text{poly}}$ measures the information density that remains after compression is restricted to efficient descriptions. The *gambling* approach defines polynomial-time dimension, denoted $\dim_{\text{P}}$, using polynomial-time betting algorithms known as s-gales [27]. Here again, an s-gale succeeds if its capital becomes arbitrarily large along the sequence, but now the gale itself must be computable in polynomial time. The value $\dim_{\text{P}}(X)$ is the infimum over all values of s for which some polynomial-time s-gale succeeds on X. Thus, $\dim_{\text{P}}$ measures the information density that cannot be exploited away by any polynomial-time betting strategy.

At the levels of polynomial-space computation [16] and finite-state computation [5, 4], these notions are known to be equivalent. At the polynomial-time level, Hitchcock and Vinodchandran proved that $\dim_{\text{P}}$ is at least as large as $\mathcal{K}_{\text{poly}}$. The converse direction has remained open for two decades. This question of *robustness of polynomial-time dimension* was posed by Hitchcock and Vinodchandran [14, 16], and later mentioned in a survey by Stull (see Open Question 3.71 in [36]).

The positive equivalences at the computability and polynomial-space levels rely on computational power that can effectively search over descriptions or simulate betting strategies. Such conversions are not available at polynomial time. Thus, in the polynomial-time setting, the robustness problem becomes sensitive to the computational limits imposed on the compression and betting algorithms. A related positive characterization is known for polynomial-time dimension when the compression side is formulated using reversible polynomial-time compressors that do not start from scratch [26]. In contrast, the question addressed in this paper concerns the original time-bounded Kolmogorov-complexity formulation of Hitchcock and Vinodchandran.

In this work, we give a surprising negative resolution to this question under the standard cryptographic assumption that one-way functions exist. First, assuming one-way functions, we construct a polynomial-time samplable distribution ν over Σ^∞ for which a set of sequences of ν-measure 1 exhibits a strict dimension gap between $\dim_{\text{P}}$ and $\mathcal{K}_{\text{poly}}$. We then strengthen this from sets to individual sequences, showing that there are sequences for which $\dim_{\text{P}} \neq \mathcal{K}_{\text{poly}}$ under the same assumption. Moreover, the gap $\dim_{\text{P}} - \mathcal{K}_{\text{poly}}$ can be made arbitrarily close to 1. The analogous non-robustness also holds for *strong* polynomial-time dimension, where lower asymptotic rates are replaced by upper asymptotic rates.

We also prove a partial converse: the existence of a polynomial-time samplable distribution supporting such dimension gaps implies the existence of infinitely-often one-way functions. This shows that the connection is not limited to deriving dimension gaps from cryptographic assumptions. Via the converse implication, dimension gaps over efficiently samplable sources lead to infinitely-often one-way functions.

Formally connecting one-way functions to the failure of robustness for these asymptotic measures of randomness requires arguments tailored to infinite sequences. The classical connections between compressibility, indistinguishability, and pseudorandomness in the finite-string setting [17, 38, 21] do not by themselves control information rates along infinitely many prefixes, or the success of a betting strategy along an infinite sequence.

The forward implication has to pass from asymptotic betting success on infinite sequences to finite-length distinguishers. A key step is an argument involving the Borel–Cantelli lemma, which isolates infinitely many lengths where the constructed source produces prefixes

on which the betting strategy has substantial gain. This is paired with a martingale estimate based on Kolmogorov's inequality, showing that comparable gains are unlikely under uniform randomness. The separation between these two behaviors is then turned into a pseudorandomness distinguisher.

The converse direction is technically different. Here the task is to show that an almost-sure dimension gap over an efficiently samplable source already carries cryptographic content. Assuming that infinitely-often one-way functions do not exist, one can approximate the probabilities of samplable distributions on typical inputs. The proof has to assemble these finite-prefix approximations into an efficient betting algorithm that works along the infinite sequence. This converts the absence of cryptographic hardness into a robustness statement for polynomial-time information density, and hence forces any such dimension gap to imply infinitely-often one-way functions.

In addition to resolving a longstanding open problem under the assumption that one-way functions exist, our results show that the robustness of polynomial-time information density is not merely a technical question internal to resource-bounded dimension. It is governed by the same phenomenon that underlies modern cryptography: the existence of efficiently samplable objects whose structure cannot be uniformly exposed by feasible computation. Thus the paper establishes a direct bridge between one-way functions and asymptotic information measures.

2 Background: Betting and Compression

Notation. Let Σ denote the binary alphabet $\{0,1\}$. Σ^n denotes the set of n-length strings and Σ^* denotes the set of all finite strings over Σ. For a finite string $w \in \Sigma^*$, $|w|$ denotes the length of w. 1^* denotes the set of all finite strings in the unary alphabet $\{1\}$. Σ^∞ denotes the *Cantor space*, or the set of all infinite strings over Σ. We use small letters w, x to represent finite strings and capital letters X, Y to represent infinite strings over Σ. For an infinite string $X \in \Sigma^\infty$, $X[i]$ denotes the i^{th} bit of X and $X \upharpoonright n$ denotes the first n bits of X. For infinite string $X = X_1X_2X_3X_4\ldots$ and $m \geq n$, $X[n:m]$ denotes the the substring $X_nX_{n+1}X_{n+2}\ldots X_m$. For a finite string $w \in \Sigma^*$, $w[i]$, $w \upharpoonright n$ and $w[n:m]$ are defined similarly. λ denotes the empty string. $\mathbb{N}$ denotes the set of natural numbers and $\mathbb{Q}$ denotes the set of rationals. All the logarithms in the paper are taken to base 2 unless specified otherwise. For any function $f : \Sigma^\infty \to \Sigma^\infty$ and $\mathcal{S} \subseteq \Sigma^\infty$ let $f(\mathcal{S})$ denote $\{f(X) : X \in \mathcal{S}\}$.

For $w \in \Sigma^*$, C_w denotes the *cylinder set* of w, that is the set of all infinite sequences in Σ^∞ that begin with w. Let $\mathcal{B}(\Sigma^\infty)$ denote the Borel σ-algebra over Σ^∞ generated by the set of all cylinder sets $\{C_w : w \in \Sigma^*\}$. For two finite strings $x, y \in \Sigma^*$, $x.y$ denotes the concatenation of the strings x and y.

We use $\text{poly}(n)$ to denote the set of polynomial (time) functions $\bigcup_{c\in\mathbb{N}}\{n^c\}$. When the context is clear, we also use poly to denote $\text{poly}(n)$. We use U_n to denote the uniform probability distribution over Σ^n.

2.1 s-gales and Kolmogorov Complexity

***s*-gales** are real-valued betting strategies on infinite sequences. An s-gale may be viewed as a gambler that reads a sequence bit by bit and repeatedly reallocates its current capital between the two possible next bits. The gale d starts with initial capital 1 on the empty string. At the n^{th} stage, after seeing a prefix $X_1X_2\ldots X_n$, the gale has capital $d(X_1X_2\ldots X_n)$ and places a bet on the next bit X_{n+1}. The rule of the game is that, for every finite string w, the capital assigned to the two possible extensions satisfies $2^s d(w) = d(w0) + d(w1)$. A gale d succeeds on the sequence $X \in \Sigma^\infty$ if it can make its capital arbitrarily large along the sequence, more precisely if $\limsup_{n\to\infty} d(X_1X_2\ldots X_n) = \infty$.

▶ **Definition 1** ([27, 28]). *For $s \in [0,\infty)$, an* s-supergale *is a function $d : \Sigma^* \to [0,\infty)$ such that $d(\lambda) < \infty$ and for every finite string $w \in \Sigma^*$, $d(w) \geq 2^{-s}(d(w0) + d(w1))$. An* s-gale *is a function $d : \Sigma^* \to [0,\infty)$ with $d(\lambda) < \infty$ for every finite string $w \in \Sigma^*$, $d(w) = 2^{-s}(d(w0) + d(w1))$. A* supermartingale *is a 1-supergale. A* martingale *is a 1-gale.*

Unless specified otherwise, for every s-supergale d, we assume that the initial capital $d(\lambda) = 1$.

▶ **Definition 2** ([27, 28],[1]). *The* success set *of an s-supergale d is $S^\infty(d) = \{ X \in \Sigma^\infty \mid \limsup_{n\to\infty} d(X \upharpoonright n) = \infty \}$. The* strong success set *of an s-supergale d is $S^\infty_{\text{str}}(d) = \{ X \in \Sigma^\infty \mid \liminf_{n\to\infty} d(X \upharpoonright n) = \infty \}$.*

For example, fix $s > 0$. The strategy that always bets all its capital on the next bit being 0 is given by $d(\lambda) = 1$, $d(w0) = 2^s d(w)$, and $d(w1) = 0$. This satisfies the s-gale condition. Along the sequence $000\cdots$, its capital after n steps is 2^{sn}, and hence it succeeds. On the other hand, a single bit 1 makes the capital drop to 0.

A more interesting example is obtained by betting according to a fixed bias. Let $0 < p < 1$, and define

$$d(w) = 2^{s|w|} p^{\#_0(w)} (1-p)^{\#_1(w)},$$

where $\#_0(w)$ and $\#_1(w)$ are the numbers of zeros and ones in w. Then $d(w0)+d(w1) = 2^s d(w)$, so d is an s-gale. If a sequence has limiting frequency p of zeros and $1-p$ of ones, then along its prefixes the exponential growth rate of this capital is $s - H(p)$, where $H(p)$ is the binary entropy. Thus, for every $s > H(p)$, this gale succeeds on such a biased sequence. This example illustrates how an s-gale can exploit statistical regularity rather than merely a fixed deterministic pattern.

When $s = 1$, the average capital after betting on $w0$ or $w1$ is equal to the previous capital $d(w)$. In this case the betting strategy is fair, and such s-gales are precisely martingales. When $s < 1$, the average capital after the bet is strictly smaller than $d(w)$. Thus the gambler is playing in an unfavorable market. This is often interpreted as "betting under inflation": unless the gambler detects structure in the sequence and bets accordingly, its capital tends to shrink. As s decreases, success requires increasingly strong regularities in the sequence.

The **Kolmogorov complexity** of a finite string x, denoted by $K(x)$, is the length of the shortest program that outputs x. That is, $K(x) = \min\{|\pi| : \mathcal{U}(\pi) = x\}$, where π is a program and $\mathcal{U}$ is the universal Turing machine. Any string x can be produced by a program that simply prints x, and therefore $K(x) \leq |x| + O(1)$. However, if the string contains regularity, there may be a much shorter program that outputs it. For instance, if all the even bits of the string are 0, then the program only needs to encode the bits at the odd indices and insert zeros in the even positions. In that case, $K(x)$ is at most about half the length of the string, up to additive constants.

▶ **Definition 3** (t-time-bounded Kolmogorov complexity [22, 24]). *Let t be a time-constructible function, and $x \in \Sigma^*$ be a finite binary string. The t-time bounded Kolmogorov complexity of x is $K_t(x) = \min\{|\Pi| \mid \Pi \in \Sigma^*$ and $\mathcal{U}(\Pi) = x$ in $t(|x|)$ steps$\}$, where $\mathcal{U}$ is any fixed efficient universal Turing machine.*

Since in the study of resource-bounded dimension we divide $K_t(x)$ by $|x|$, our results are unaffected by the choice between prefix-free Kolmogorov complexity and plain Kolmogorov complexity. We use plain Kolmogorov complexity in the rest of the paper.

2.2 Polynomial-time Dimension

Resource-bounded dimension is defined by placing resource bounds on the computation of the s-gale. Analogously, resource-bounded Kolmogorov complexity is defined by placing resource bounds on the programs that print a string. **Polynomial-time dimension** quantifies the rate of information in an infinite sequence, measured with respect to polynomial-time bounded computation. It is formulated using polynomial-time s-gales. An s-gale d wins on an infinite sequence X if $\limsup_{n\to\infty} d(X \upharpoonright n) = \infty$. The gale d is said to be polynomial-time computable if, for some $p(n) \in \mathrm{poly}(n)$, the value $d(w)$ can be computed in at most $p(n)$ time, where n is the length of w. Intuitively, $\dim_{\mathrm{P}}(X)$ is the infimum of the values of s for which there exists a polynomial-time s-gale that succeeds on X. Smaller values of s correspond to a more unfavorable betting environment. Thus, a low polynomial-time dimension means that some efficient betting strategy can exploit substantial regularity in the sequence, whereas high polynomial-time dimension means that the sequence resists such efficient betting.

On the other hand, in the *incompressibility approach*, we use the *time-bounded Kolmogorov complexity* to define a polynomial-time analogue of $\mathcal{K}$. For a given time bound $t(n)$, the ***t*-time bounded Kolmogorov complexity** $K_t(x)$ of a string x is the length of the shortest program that generates x in at most $t(|x|)$ time steps (see [24]).

For a given time bound $t(n)$, the quantity $\liminf_{n\to\infty} K_t(X \upharpoonright n)/n$ gives the asymptotic t-time-bounded rate of information in an infinite string X. The polynomial-time analogue of $\mathcal{K}$ is the infimum of this quantity over all polynomial-time bounds t.

Now, we define the asymptotic polynomial-time density of information of a subset $\mathcal{F} \subseteq \Sigma^\infty$.

▶ **Definition 4** ($\mathcal{K}_{\mathrm{poly}}$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ [16, 36, 15]). *For any $\mathcal{F} \subseteq \Sigma^\infty$, define $\mathcal{K}_{\mathrm{poly}}(\mathcal{F}) = \inf_{t\in\mathrm{poly}} \sup_{X\in\mathcal{F}} \liminf_{n\to\infty} \frac{K_t(X\upharpoonright n)}{n}$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(\mathcal{F}) = \inf_{t\in\mathrm{poly}} \sup_{X\in\mathcal{F}} \limsup_{n\to\infty} \frac{K_t(X\upharpoonright n)}{n}$.*

For individual sequences, we define $\mathcal{K}_{\mathrm{poly}}(X) = \mathcal{K}_{\mathrm{poly}}(\{X\})$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) = \mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(\{X\})$. Note that $\mathcal{K}_{\mathrm{poly}}$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ are equivalently defined in the following way.

▶ **Definition 5** ($\mathcal{K}_{\mathrm{poly}}$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ [16, 36, 15]). *For any $X \in \Sigma^\infty$, define $\mathcal{K}_{\mathrm{poly}}(X) = \inf_{t\in\mathrm{poly}} \liminf_{n\to\infty} \frac{K_t(X\upharpoonright n)}{n}$ and $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) = \inf_{t\in\mathrm{poly}} \limsup_{n\to\infty} \frac{K_t(X\upharpoonright n)}{n}$.*

Now, we define time-bounded supergales. We first define $t(n)$-time computable s-gales.

▶ **Definition 6** ($t(n)$-time computability [27]). *An s-supergale $d : \Sigma^* \to [0,\infty)\cap\mathbb{Q}$ is $t(n)$-time computable if there exist a $O(t(n))$-time computable function $f : \Sigma^* \times 1^* \to \mathbb{Q}$ such that $|d(w) - f(w, 1^n)| \leq 2^{-n}$ for every $w \in \Sigma^*$ and $n \in \mathbb{N}$.*

Another notion of time-bounded computability is that of exact $t(n)$-time computability.

▶ **Definition 7** (Exact $t(n)$-time computability [27]). *An s-supergale $d : \Sigma^* \to [0,\infty) \cap \mathbb{Q}$ is exact $t(n)$-time computable if there exist a $O(t(n))$-time computable function $f : \Sigma^* \to \mathbb{Q}$ such that $d(w) = f(w)$ for every $w \in \Sigma^*$.*

Now, we define polynomial-time dimension and strong dimension.

▶ **Definition 8** (Polynomial-time dimension [27] and strong dimension [1, 36]). *The polynomial-time dimension of $\mathcal{F} \subseteq \Sigma^\infty$ is defined as*

$$\dim_{\mathrm{P}}(\mathcal{F}) = \inf\{s \mid (\exists k) \text{ there is an exact } n^k\text{-time computable } s\text{-gale } d \text{ such that } \mathcal{F} \subseteq S^\infty(d)\}.$$

For a sequence $X \in \Sigma^\infty$*, define* $\dim_{\mathrm{P}}(X) = \dim_{\mathrm{P}}(\{X\})$*. The* polynomial-time strong dimension $\mathrm{Dim}_{\mathrm{P}}(\mathcal{F})$ *of* $\mathcal{F} \subseteq \Sigma^\infty$ *is defined by replacing* S^∞ *by* S^∞_{str} *in the above expression. For a sequence* $X \in \Sigma^\infty$*, define* $\mathrm{Dim}_{\mathrm{P}}(X) = \mathrm{Dim}_{\mathrm{P}}(\{X\})$.

We now make an important remark. Polynomial-time dimension and strong dimension are often defined in terms of n^k-time computable s-gales that need not be exact computable. But these definitions are in fact equivalent. This is a direct consequence of the following lemma from [27] which is also useful in proving the main results in this paper.

▶ **Lemma 9** (Exact Computation Lemma [27]). *If there is a polynomial-time computable s-gale d such that $2^s \in \mathbb{Q}$, then there exists an exact polynomial-time computable s-gale $\tilde{d}$ such that $S^\infty(d) \subseteq S^\infty(\tilde{d})$.*

We will often use the terminology polynomial-time computable gales to refer to exact $t(n)$-time computable s-gales where $t(n) \in \mathrm{poly}(n)$. Similar comment applies in the case of exact polynomial-time computability and other notions related to gales such as martingales and supergales.

2.3 Robustness of Dimension

At the computability level, the relevant notions are $\mathcal{K}$, defined using time-unbounded Kolmogorov complexity $K(x)$, and dim, defined using lower semicomputable s-gales. Lutz [28] and Mayordomo [29] showed that the unbounded notions of $\mathcal{K}$ and dim are equivalent, hence establishing the robustness of *constructive* dimension.

PSPACE analogues of dim and $\mathcal{K}$ are defined by restricting the s-gales in the definition of dim to be polynomial-space computable and by using polynomial-space bounded Kolmogorov complexity in the definition of $\mathcal{K}$. Hitchcock and Vinodchandran [14, 16] showed that these notions of resource-bounded dimension coincide in the PSPACE setting. Thus polynomial-space dimension is robust in the same sense.

This leads to the following question: *Are the notions of polynomial-time dimension formulated using s-gales ($\dim_{\mathrm{P}}$) and time-bounded Kolmogorov complexity ($\mathcal{K}_{\mathrm{poly}}$) equivalent?* Hitchcock and Vinodchandran [16] showed that $\dim_{\mathrm{P}}$ is always greater than or equal to $\mathcal{K}_{\mathrm{poly}}$.

▶ **Theorem 10** ([14, 16]). *For every $\mathcal{F} \subseteq \Sigma^\infty$, $\mathcal{K}_{\mathrm{poly}}(\mathcal{F}) \leq \dim_{\mathrm{P}}(\mathcal{F})$.*

The question whether the reverse inequality holds has remained elusive. Hitchcock and Vinodchandran [14, 16] and later Stull in [36] posed the following open question:

▶ **Question 11** ([14, 16, 36]). *Is it true that, for every sequence $X \in \Sigma^\infty$, $\dim_{\mathrm{P}}(X) = \mathcal{K}_{\mathrm{poly}}(X)$?*

Polynomial-time strong dimension is a dual notion of polynomial-time dimension [1, 36]. As in the case of polynomial-time dimension, there exist two notions, one defined using s-gales and the other defined in terms of time-bounded Kolmogorov complexity. An s-gale d strongly succeeds on an infinite sequence X if $\liminf_{n\to\infty} d(X_1X_2\ldots X_n) = \infty$.

The strong dimension analogue of $\mathcal{K}_{\mathrm{poly}}$ is defined by replacing the lim inf in the definition of $\mathcal{K}_{\mathrm{poly}}$ with lim sup. The analogue of Theorem 10 also holds in this setting: $\mathrm{Dim}_{\mathrm{P}}$ is at least as large as $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$. Stull in [36] posed the following question for the dual notion of strong dimension:

▶ **Question 12.** *Are the notions of polynomial-time strong dimension formulated using s-gales ($\mathrm{Dim}_{\mathrm{P}}$) and time-bounded Kolmogorov complexity ($\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$) equivalent for every sequence $X \in \Sigma^\infty$?*

We answer both questions in the negative, under the standard cryptographic assumption that one-way functions exist.

2.4 Measure over Cantor Space

▶ **Definition 13** ([34]). *A function $\mu : \mathcal{M} \to [0, \infty]$, where $\mathcal{M} \subseteq P(\Sigma^\infty)$, is called a measure over $(\Sigma^\infty, \mathcal{M})$ if $\mu(\emptyset) = 0$ and μ is countably additive. That is, for any countable disjoint collection $\{E_k\}_{k=1}^\infty$ of measurable sets, $\mu\left(\bigcup_{k=1}^\infty E_k\right) = \sum_{k=1}^\infty \mu(E_k)$.*

We take $\mathcal{M}$ as the set of Borel Measurable sets. Note that $\mathcal{B} \subseteq \mathcal{M}$, where $\mathcal{B}$ is the Borel sigma algebra over Σ^∞, we omit $\mathcal{M}$ and say that μ is a measure over Σ^∞.

▶ **Definition 14.** *Let ν be a measure over Σ^∞. We say that ν is a probability distribution over Σ^∞ if it holds that $\nu(\Sigma^\infty) = 1$.*

Let ν be a probability distribution over Σ^∞. For every n, let ν_n denote the probability distribution induced by ν on Σ^n defined as $\nu_n(w) = \nu(C_w)$ for every $w \in \Sigma^*$.

Let Σ^∞ be the Cantor space. Now we define the uniform (Lebesgue) measure over the Cantor space [7, 31]. The uniform measure of a cylinder set C_w determined by a finite string w is defined as: $\mu(C_w) = 2^{-|w|}$. Using routine measure theoretic arguments, μ uniquely extends to a probability measure over $(\Sigma^\infty, \mathcal{B}(\Sigma^\infty))$ (see [2]).

2.5 One-way Functions and Pseudorandom Generators

One-way functions [6, 10, 23] are functions that are easy to compute but are hard to invert, except possibly on a negligible fraction of inputs. They are central in cryptography, since their existence is necessary for that of essential cryptographic primitives [17, 33, 12, 8, 30].

The following is the definition of one-way functions secure against uniform probabilistic polynomial-time adversaries.

▶ **Definition 15** (One-way functions [11, 37, 10]). *A function $f_n : \Sigma^* \to \Sigma^*$ is a one-way function (OWF) if there is a constant b such that f_n is computable in time n^b for sufficiently large n, and for every probabilistic polynomial-time algorithm A and every constant $c > 0$, $\Pr[A(f_n(U_n)) \in f_n^{-1}(f_n(U_n))] \leq \frac{1}{n^c}$, for all sufficiently large n, where the probability is taken over the choice of U_n and the internal randomness of algorithm A.*

To define pseudorandom generators, we first define computational indistinguishability.

▶ **Definition 16** (Computational Indistinguishability [11, 37]). *The ensembles of random variables $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$ are computationally indistinguishable if for every probabilistic polynomial-time algorithm T and every constant $c > 0$,*

$$|\Pr[T(X_n) = 1] - \Pr[T(Y_n) = 1]| \leq \frac{1}{n^c}$$

where the probabilities are taken over the distributions of either X_n or Y_n and the internal randomness of the algorithm T.

▶ **Definition 17** (Pseudorandom generators [11, 37, 10]). *A sequence of functions $\{G_n : \Sigma^{d(n)} \to \Sigma^n\}_{n\in\mathbb{N}}$ is a pseudorandom generator (PRG) if $d(n) < n$ for all n, there exists a uniform deterministic polynomial-time algorithm M such that $M(n, x) = G_n(x)$, and the ensembles $\{G_n(U_{d(n)})\}_{n\in\mathbb{N}}$ and $\{U_n\}_{n\in\mathbb{N}}$ are computationally indistinguishable.*

The theorem below states the equivalence of one-way functions and pseudorandom generators.

▶ **Theorem 18** (Equivalence theorem [17, 11, 10]). *The following are equivalent.*

1. *One-way functions exist.*
2. *There exist a pseudorandom generator with seed length $d(m) = m - 1$.*
3. *For every polynomial-time computable constant $\epsilon \in (0, 1)$, there exists a pseudorandom generator with seed length $d(m) = \epsilon \cdot m$.*

2.6 Efficiently Samplable Sources and Approximable Gales

We generalize the notion of polynomial-time samplable distributions [3] to probability distributions on the space of infinite sequences. This is the distributional setting in which our main duality theorem is stated: instead of considering arbitrary sequences alone, we also study typical outputs of efficiently samplable sources over Σ^∞.

Let ν be a probability measure over Σ^∞. For every n, let $\nu_n(w) = \nu(X \in \Sigma^\infty : w \sqsubseteq X)$[1] denote the probability distribution induced by ν on Σ^n. Now, we define polynomial-time samplable distributions on Σ^∞.

▶ **Definition 19.** *A probability distribution ν over Σ^∞ is polynomial-time samplable if there exists a probabilistic Turing machine M that uses $q(n)$ random bits, where q is a polynomial, such that for every n and $w \in \Sigma^n$, $\Pr_{r \sim \Sigma^{q(n)}}[M(1^n, r) = w] = \nu_n(w)$.*

An alternate way of defining polynomial-time samplable distributions on Σ^∞ is the following: $\nu = \{\nu_n\}$ is a *polynomial-time samplable distribution on* Σ^∞ if $\{\nu_n\}$ is a *polynomial-time samplable distribution* ([3]) such that for any $w \in \Sigma^n$, $\nu_n(w) = \nu_{n+1}(w.0) + \nu_{n+1}(w.1)$. The equivalence of these notions follows using routine measure theoretic arguments.

Primitives like one-way functions and pseudorandom generators are defined against *probabilistic* polynomial-time adversaries, whereas polynomial-time s-gales are *deterministic* betting strategies. To compare these two kinds of objects, we use a mild average-case evaluation notion for gales. The betting strategy itself remains a deterministic s-supergale, with the usual success condition along infinite sequences. The relaxation is only in how its capital function is accessed: on inputs drawn from the target samplable distribution, the capital is required to have an efficient randomized constant-factor approximation with high probability.

This is the natural interface for our distributional arguments. On the cryptographic side, pseudorandomness and probability-estimation results are inherently probabilistic and average-case over samples from a samplable distribution. On the dimension side, the object being analyzed is still a gale. Exact polynomial-time computable gales are included as a special case, since their values can be returned deterministically. Thus the notion below does not replace polynomial-time dimension. It is a technical strengthening used to formulate the almost-sure distributional separation and its converse.

This is in the spirit of the probabilistic approximation notions for martingales studied by Regan and Sivakumar [32], but we only need an inverse-polynomial average-case guarantee over ν_n: for each length n and every constant k, the estimator approximates $d(w)$ within a fixed constant factor on all but an n^{-k} fraction of samples $w \sim \nu_n$, and any approximation errors are allowed only on strings in support of ν_n.

▶ **Definition 20.** *Let $d : \Sigma^* \to [0, \infty) \cap \mathbb{Q}$ be an s-supergale and ν be any probability distribution over Σ^∞. d is $t(n)$-time ν-approximable if for every constant k there exist a probabilistic $t(n)$-time machine M and constant $c < 1$ such that for every n, $\{w \in \Sigma^n : M(w) \notin [c \cdot d(w), d(w)]\} \subseteq \mathrm{supp}(\nu_n)$ and $\nu_n\{w \in \Sigma^n : M(w) \notin [c \cdot d(w), d(w)]\} \leq n^{-k}$.*

[1] $\sqsubseteq$ denotes the prefix operator

Here $\mathrm{supp}(\nu_n) = \{w \in \Sigma^n : \nu_n(w) > 0\}$. The support condition prevents the approximation procedure from hiding errors on impossible prefixes. Equivalently, outside the support of the sampled prefix distribution, the estimator must still output a value within the specified multiplicative range, while on typical samples from the source it may fail only with inverse-polynomial probability.

3 Our Results

3.1 One-way functions and dimension gaps

Our main result gives a two-sided connection between one-way functions and dimension gaps over efficiently samplable sources. Assuming one-way functions, we construct a polynomial-time samplable distribution over Σ^∞ whose typical sequences have low polynomial-time Kolmogorov complexity rate but cannot be captured by low-dimensional polynomial-time betting strategies. Conversely, the existence of such a samplable source already implies infinitely-often one-way functions.

▶ **Theorem 21.** *In the following, (1) $\Longrightarrow$ (2) $\Longrightarrow$ (3):*

1. *One-way functions exist.*
2. *For every $s < 1/2$, there exists a polynomial-time samplable distribution ν over Σ^∞ such that:*
   a. *$\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \le s$ for almost every $X \sim \nu$.*
   b. *For every $s' \in (s, 1/2)$ and every polynomial-time ν-approximable s'-supergale d, we have $\nu(S^\infty(d)) = 0$.*
3. *Infinitely-often one-way functions exist.*

Item (2a) says that, with ν-probability 1, the sampled sequence has small polynomial-time Kolmogorov complexity rate. In particular, $\mathcal{K}_{\mathrm{poly}}(X) \le s$ for ν-almost every $X \sim \nu$. The second part of Item (2) rules out success by low-dimensional polynomial-time ν-approximable supergales. Lemma 42 shows that, under the same assumption, this yields $\dim_{\mathrm{P}}(X) \ge 1/2$ for ν-almost every X. Since $s < 1/2$, this gives an almost-sure dimension gap over the samplable source.

It is important that the approximation notion in Item (2b) does not change the polynomial-time dimension appearing in the applications below. Every exact polynomial-time computable gale is polynomial-time ν-approximable, by returning its value deterministically. Thus Item (2b) rules out, in particular, ordinary polynomial-time betting strategies on positive-measure subsets of the samplable source. The approximability condition is used only to state the distributional theorem in a form compatible with pseudorandomness arguments and average-case probability-estimation results.

We prove Theorem 21 in two steps. First we show that one-way functions yield a polynomial-time samplable distribution exhibiting an almost-everywhere dimension gap between $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ and $\dim_{\mathrm{P}}$ (Lemma 25, giving (1)⇒(2)), and then we show that the existence of such a dimension-gap distribution yields infinitely-often one-way functions (Lemma 36, giving (2)⇒(3)).

3.2 Applications

Theorem 21 resolves the original robustness question for individual sequences, assuming one-way functions.

▶ **Theorem 22.** *If for all $X \in \Sigma^\infty$, $\dim_{\mathrm{P}}(X) = \mathcal{K}_{\mathrm{poly}}(X)$, then no one-way functions exist.*

Equivalently, assuming one-way functions, there are individual sequences X such that $\mathcal{K}_{\text{poly}}(X) < \dim_{\text{P}}(X)$. We extend this result to show that if one-way functions exist, then the distance between these quantities can be arbitrarily close to the maximum possible value of 1.

▶ **Theorem 23.** *If one-way functions exist, then for any $\epsilon > 0$, there exists $X \in \Sigma^\infty$ such that $\dim_{\text{P}}(X) - \mathcal{K}_{\text{poly}}(X) \geq 1 - \epsilon$.*

For polynomial-time strong dimension, we show that if one-way functions exist, then there exist sequences for which the gap between $\mathcal{K}^{\text{str}}_{\text{poly}}$ and Dim_{P} is arbitrarily close to 1.

▶ **Theorem 24.** *If one-way functions exist, then for any $\epsilon > 0$, there exists $X \in \Sigma^\infty$ such that $\text{Dim}_{\text{P}}(X) - \mathcal{K}^{\text{str}}_{\text{poly}}(X) \geq 1 - \epsilon$.*

Therefore, we show that if one-way functions exist, the answers to both parts of Open Question 3.71 from [36] are negative.

4 Proof Outline and Techniques

In this section we describe the main constructions and reductions behind our results. The emphasis is on how finite-length pseudorandomness arguments interact with asymptotic information-density notions over infinite sequences.

The main difficulty is that the two sides are formally quite different. Pseudorandomness is a finite-length indistinguishability notion against probabilistic polynomial-time tests, while dimension is an asymptotic notion defined by success of betting strategies along infinitely many prefixes of one sequence. The proof has to convert information in both directions: from asymptotic betting success to finite distinguishers, and from average-case probability estimation for samplable distributions to betting objects on infinite sequences.

4.1 From one-way functions to dimension gaps

We first show how one-way functions, through pseudorandom generators, yield efficiently samplable sources with an almost-everywhere dimension gap. Concretely, assuming one-way functions, for every $s < 1/2$ we construct a polynomial-time samplable distribution ν over Σ^∞ such that (i) $\mathcal{K}^{\text{str}}_{\text{poly}}(X) \leq s$ for ν-almost every X, and (ii) for every $s' \in (s, 1/2)$, no polynomial-time ν-approximable s'-supergale succeeds on a set of positive ν-measure.

▶ **Lemma 25.** *If one-way functions exist, then for every $s < 1/2$, there exist a polynomial-time samplable distribution ν over Σ^∞ such that:*

1. *$\mathcal{K}^{\text{str}}_{\text{poly}}(X) \leq s$ for almost every $X \sim \nu$.*
2. *For every $s' \in (s, 1/2)$ and any polynomial-time ν-approximable s'-supergale d, $\nu(S^\infty(d)) = 0$.*

Fix $s < 1/2$. Assume one-way functions exist, and choose a dyadic parameter $s_0 = 2^{-m}$ with $2s_0 < s$. Then there are pseudorandom generators $\{G_n\}_{n\in\mathbb{N}}$ running in time $t_G(n) \in \text{poly}(n)$ with constant stretch: for each n, $G_n : \Sigma^{s_0 n} \to \Sigma^n$. The choice of such an s_0 will give the stronger compression bound needed to obtain $\mathcal{K}^{\text{str}}_{\text{poly}}(X) \leq s$.

4.1.1 Construction of the PRG-induced source

We extend the family $\{G_n\}$ to a mapping on infinite sequences, $g : \Sigma^\infty \to \Sigma^\infty$, by applying the appropriate G_{2^i} blockwise to successive blocks of the input. Given a PRG $G_n : \Sigma^{s_0 n} \to \Sigma^n$ where $s_0 = 2^{-m}$ for some $m \in \mathbb{N}$, we construct a mapping g from the Cantor set to itself, by

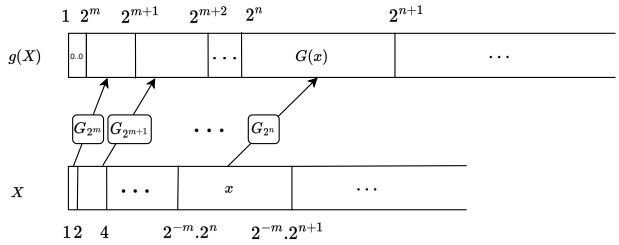

Figure 1 Illustration of the construction of $g(X)$. The first 2^m bits of $g(X)$ are 0. Thereafter, for every $n > m$, the block $X[s_0 \cdot 2^{n-1}, s_0 \cdot 2^n - 1]$ is mapped to $G_{2^{n-1}}(X[s_0 \cdot 2^{n-1}, s_0 \cdot 2^n - 1])$ in the output string $g(X)$.

using the family $\{G_n\}_{n\in\mathbb{N}}$ to map blocks within the input string to *longer* blocks within the output string. Given an input $X \in \Sigma^\infty$, $g(X)$ is constructed by applying G_n to successive blocks x of size $s_0 \cdot 2^n$ from X for all $n \in \mathbb{N}$. We first define a sequence of mappings g_k from Σ^k to Σ^*. These mappings preserve the prefix ordering among finite strings and hence we define $g(X)$ for $X \in \Sigma^\infty$ as the limit of $g_k(X \upharpoonright k)$.

▶ **Definition 26.** *Let $G_n : \Sigma^{s_0 n} \to \Sigma^n$ where $s_0 = 2^{-m}$ for some $m \in \mathbb{N}$ and let $k \in \mathbb{N}$. For $w \in \Sigma^k$, define $g_k(w)$ as follows. For any n satisfying any $m < n \leq \lfloor \log(k/s_0) \rfloor$ and $i \in \{2^{n-1}+1, 2^{n-1}+2, \ldots 2^n\}$, let $g_k(w)[i] = G_{2^{n-1}}(x)[i-2^{n-1}]$, where $x = w[s_0 \cdot 2^{n-1}, s_0 \cdot 2^n - 1]$. For $i \in \{1, 2, \ldots, 2^m\}$, we set $g_k(w)[i] = 0$.*

Note that $g_k(w)[2^{n-1}+1, 2^n] = G_{2^{n-1}}(w[s_0 \cdot 2^{n-1}, s_0 \cdot 2^n - 1])$. For $k \leq 2^m$, g_k maps any $w \in \Sigma^k$ to 0^{2^m}. Observe that the sequence of mappings $\langle g_k \rangle_{k\in\mathbb{N}}$ preserves the prefix order between strings. That is, If $w' \sqsubseteq w$, then $g_{|w'|}(w') \sqsubseteq g_{|w|}(w)$. Therefore, we define g as the limit of the mappings g_k as $k \to \infty$.

▶ **Definition 27.** *For every $X \in \Sigma^\infty$ let $g(X) = \cap_{n=1}^\infty C_{g(X\upharpoonright n)}$.*

Notice that for any $X \in \Sigma^\infty$ and $k \in \mathbb{N}$, $g_k(X \upharpoonright k) \sqsubseteq g(X)$. For convenience, we abuse the notation and use g to denote the mapping between infinite strings defined above or a mapping between finite strings defined as follows: $g(w) = g_{|w|}(w)$ for any $w \in \Sigma^*$. The intended meaning is clear from the context.

The mapping g induces a polynomial-time samplable distribution ν over Σ^∞. The sampling algorithm M for ν on input 1^n maps a random seed of length $s_0 \cdot 2^{\lceil \log_2(n) \rceil}$ using g to a string of length $2^{\lceil \log_2(n) \rceil}$, and thereafter trims the output to n bits. The number of random bits used by M on input 1^n is at most $2s_0 \cdot n$. Intuitively, because the sampler uses at most $2s_0 n$ random bits on input 1^n, those random bits, together with a fixed $O(1)$-bit description of the sampler, provide a short polynomial-time description of the output prefix. Thus, for ν-a.e. X, we obtain $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \leq 2s_0 < s$. The formal complexity bound is given by the following lemma and corollary.

▶ **Lemma 28.** *For $s_0 = 2^{-m}$ for some $m > 1$, let $G_n : \Sigma^{s_0 n} \to \Sigma^n$ be a PRG that runs in time $t_G(n)$. Let $g : \Sigma^\infty \to \Sigma^\infty$ be the function constructed from G_n as in Definition 26. There exists a constant $c \in \mathbb{N}$ such that for all $X \in \Sigma^\infty$ and for all $n \in \mathbb{N}$, $K_{t_g}(g(X) \upharpoonright 2^n) \leq s_0 \cdot 2^n + c$, where $t_g(n) = t_G(n) \cdot n$.*

Proof. Consider the algorithm M which implements the mapping in Definition 26 as follows.

Algorithm 1 Algorithm M.

Input: A string w.
if $w = \lambda$ **then**
    Output λ
else
    Output 0^{2^m}
end if
for $i = m$ to $\lfloor \log(|w|2^m) \rfloor - 1$ **do**
    Output $G_{2^i}(w[2^{-m} \cdot 2^i, 2^{-m} \cdot 2^{i+1} - 1])$
end for

Since G_n runs in $t_G(n)$ time, it follows that M runs in $t_g(n) = t_G(n) \cdot n$ time. Also, for any $X \in \Sigma^\infty$ and $n \geq m$, M outputs the string $g(X) \upharpoonright 2^n$ on input $X \upharpoonright 2^{-m} \cdot 2^n = X \upharpoonright s_0 \cdot 2^n$. Therefore, it follows that $K_{t_g}(g(X) \upharpoonright 2^n) \leq s_0 \cdot 2^n + c$ where the constant c depends only on the length of any representation of M in the prefix language of the fixed universal Turing machine. ◀

We obtain the following corollary.

▶ **Corollary 29.** *For $s_0 = 2^{-m}$ for some $m > 1$, let $G_n : \Sigma^{s_0 n} \to \Sigma^n$ be a PRG that runs in time $t_G(n)$. Let $g : \Sigma^\infty \to \Sigma^\infty$ be the function constructed from G_n as in Definition 26. There exists a constant $c \in \mathbb{N}$ such that for all $X \in \Sigma^\infty$ and for all $i \in \mathbb{N}$, $K_{t_g}(g(X) \upharpoonright i) \leq 2s_0 \cdot i + 2\log(i) + c'$, where $t_g(n) = t_G(n)n$.*

Proof. From Lemma 28, $K_{t_g}(g(X) \upharpoonright 2^{\lceil log(i) \rceil}) \leq s_0 \cdot 2^{\lceil log(i) \rceil} + c'$. With $K_n(i)$ extra bits, we specify the index i at which $g(X) \upharpoonright 2^{\lceil log(i) \rceil}$ needs to be trimmed. Thus, we obtain a program that outputs $g(X) \upharpoonright i$ in $t_g(n)$-time. There is a trivial prefix-free program which given $2\log(i)$ bits of information, runs in linear time and prints i. Hence, the value of $K_n(i) \leq 2\log(i) + O(1)$. Since $2^{\lceil log(i) \rceil} \leq 2i$, there exists $c' > 0$ such that $K_{t_g}(g(X) \upharpoonright i) \leq 2s_0 \cdot i + K_n(i) + c \leq 2s_0 \cdot i + 2\log(i) + c'$. ◀

4.1.2 From betting gains to a finite-length distinguisher

Suppose toward a contradiction that there exist $s' \in (s, 1/2)$ and a polynomial-time ν-approximable s'-supergale d' with $\nu(S^\infty(d')) > 0$. Since $2s_0 < s < s'$, this is in particular a gale exponent above the compression rate obtained from the PRG-induced source. Using standard gale manipulations, we convert d' into a polynomial-time ν-approximable supermartingale $\tilde{d}$ that exhibits large multiplicative gains between dyadic checkpoints 2^{n-1} and 2^n infinitely often along every sequence in some positive-ν-measure set. This conversion is carried out in two stages, established in the following lemmas.

▶ **Lemma 30.** *For $s' \in [0, \infty)$ if $d' : \Sigma^* \to \mathbb{R}_{\geq 0}$ is a $t(n)$-time computable ν-approximable s'-supergale, then for any s'' satisfying $s' \leq s'' \leq 1$ and $2^{s''} \in \mathbb{Q}$, there exists an exact $t(n)poly(n)$-time ν-approximable supermartingale $\tilde{d} : \Sigma^* \to \mathbb{R}_{\geq 0}$ such that for any $X \in S^\infty(d')$,*

$$\limsup_{n \to \infty} 2^{-(1-s'')n} \tilde{d}(X \upharpoonright n) = \infty.$$

Proof. Let $s'' \geq s'$ such that $2^{s''} \in \mathbb{Q}$. We know that d' is a $t(n)$-time ν-approximable s''-supergale that succeeds on X (see Observation 4.4 from [27]). Let $\bar{d}$ be defined as $\bar{d}(w) = d'(w)2^{(1-s'')|w|}$. Note that as d' is ν-approximable, it follows that $\bar{d}$ is ν-approximable.

From Observation 3.2 in [28], it follows that $\bar{d}$ is a supermartingale such that,

$$\limsup_{n\to\infty} \frac{\bar{d}(X \upharpoonright n)}{2^{(1-s'')n}} = \infty.$$

Since $2^{s''}$ is rational, so is $2^{(1-s'')}$. Hence, $\bar{d}$ is an $t(n)poly(n)$-time ν-approximable supermartingale. ◀

▶ **Lemma 31.** *For $s' < 1/2$, let $d' : \Sigma^* \to \mathbb{R}_{\geq 0}$ be an $t(n)$-time ν-approximable s'-supergale. Let $\tilde{s}$ be such that $\tilde{s} > s'$, $2^{\tilde{s}} \in \mathbb{Q}$. Then, there exists an $t(n)poly(n)$-time ν-approximable supermartingale $\tilde{d} : \Sigma^* \to \mathbb{R}_{\geq 0}$ such that for any $X \in S^\infty(d')$, there exists infinitely many $n \in \mathbb{N}$ such that $\tilde{d}(X \upharpoonright 2^n) > 2^{(1-\tilde{s})2^{n-1}} \tilde{d}(X \upharpoonright 2^{n-1})$.*

Proof. Choose $s'' \in (s', 1/2)$ such that $s' < 2s' < 2s'' < \tilde{s}$ and $2^{s''} \in \mathbb{Q}$. Let $\tilde{d}$ be the $t(n)poly(n)$-time ν-approximable supermartingale from Lemma 30 such that

$$\limsup_{n\to\infty} \frac{\tilde{d}(X \upharpoonright n)}{2^{(1-s'')n}} = \infty.$$

Suppose, toward a contradiction, that for some $m > 0$ and every $n > m$,

$$\tilde{d}(X \upharpoonright 2^n) \leq 2^{(1-\tilde{s})2^{n-1}} \tilde{d}(X \upharpoonright 2^{n-1}).$$

Then, for any $n > m$ and $\ell \leq 2^n$,

$$\tilde{d}(X \upharpoonright (2^n + \ell)) \leq 2^\ell 2^{(1-\tilde{s})\sum_{i=m}^{n-1} 2^i} \tilde{d}(X \upharpoonright 2^m) = 2^\ell 2^{(1-\tilde{s})(2^n - 2^m)} \tilde{d}(X \upharpoonright 2^m).$$

If also $\tilde{d}(X \upharpoonright (2^n + \ell)) \geq 2^{(2^n+\ell)(1-s'')}$, then rearranging gives

$$2^{\ell s''} \geq \frac{2^{(1-\tilde{s})2^m}}{\tilde{d}(X \upharpoonright 2^m)} 2^{2^n(\tilde{s}-s'')}.$$

Since $\ell \leq 2^n$, we obtain

$$2^{2^n(\tilde{s}-2s'')} \leq \frac{\tilde{d}(X \upharpoonright 2^m)}{2^{(1-\tilde{s})2^m}}. \tag{1}$$

If there are infinitely many k such that $\tilde{d}(X \upharpoonright k) \geq 2^{(1-s'')k}$, then (1) holds for infinitely many n. But this is impossible for all large enough n, since $\tilde{s} > 2s''$. Therefore, for infinitely many n,

$$\tilde{d}(X \upharpoonright 2^n) > 2^{(1-\tilde{s})2^{n-1}} \tilde{d}(X \upharpoonright 2^{n-1}),$$

as required. ◀

Let M be a polynomial-time ν-approximator for $\tilde{d}$. We then define a polynomial-time algorithm A that, on input length 2^n, samples a random r, forms the corresponding prefix $w' = g(r) \upharpoonright 2^n$, and checks via M whether appending the input causes $\tilde{d}$ to jump by the required multiplicative factor.

4.1.3 The distinguisher algorithm

Let $\tilde{d}$ be the $t(n) \cdot poly(n)$-time ν-approximable martingale obtained from Lemma 31. Let M be the Turing machine that ν-approximates d. That is for all $n \in \mathbb{N}$, $\nu_n\{w \in \Sigma^n : M(w) \notin [\mathbf{c} \cdot d(w), d(w)]\} \leq n^{-k}$ where $\mathbf{c} \in \mathbb{Q}, k \in \mathbb{N}$ are constants such that $\mathbf{c} \leq 1$ and $k \geq 1$. We use M to build a distinguisher algorithm A that breaks the PRG $\{G_n\}$.

Algorithm 2 Algorithm A.

Input: A string w where $|w| = 2^n$
$r \leftarrow \text{random}(s_0 \cdot 2^n)$ ▷ Generate a random seed of length $s_0 \cdot 2^n$
for $i = m$ to $n-1$ **do**
    $v_i \leftarrow G_{2^i}(r[s_0 \cdot 2^i, s_0 \cdot 2^{i+1} - 1])$ ▷ Use r and G_{2^i} to get a PRG output of length 2^i
end for
$w' \leftarrow 0^{2^m} v_1 \dots v_n$ ▷ Concatenate all v_i
return 1 **if** $M(w'w) \geq \mathbf{c} \cdot 2^{(1-\tilde{s})|w|} \cdot M(w')$.

4.1.4 Performance on PRG outputs

A remaining subtlety is that the martingale guarantee is only *infinitely often* along each sequence in a positive-measure set. We upgrade this to a *noticeable fraction at infinitely many lengths* using a Borel–Cantelli argument, which yields infinitely many n for which a polynomially-large set of seeds r witnesses the required dyadic-block gain. At these lengths, the ν-approximation property of M ensures that A outputs 1 with inverse-polynomial probability on PRG outputs.

We first define the function f such that $f(X, n) = 1$ if $\tilde{d}$ has a sufficiently large gain in the n-th block of X, and $f(X, n) = 0$ otherwise.

▶ **Definition 32.** *Let $f : \Sigma^\infty \times \mathbb{N} \to \{0,1\}$ be defined by specifying $f(X,n) = 1$ if $\tilde{d}(X \upharpoonright 2^n) > 2^{(1-\tilde{s})2^n} \tilde{d}(X \upharpoonright 2^{n-1})$ and 0 otherwise.*

The following fact is a consequence of the Borel Cantelli Lemma [2].

▶ **Lemma 33.** *Let $f : \Sigma^\infty \times \mathbb{N} \to \{0,1\}$ and $\mathcal{S} \subseteq \Sigma^\infty$ such that $\nu(\mathcal{S}) > 0$. If for every $X \in \mathcal{S}$, there exist infinitely many $n \in \mathbb{N}$ such that $f(X,n) = 1$, then for any $c > 1$, there exist infinitely many $n \in \mathbb{N}$, such that $\nu(\{X : f(X,n) = 1\}) > n^{-c}$.*

Now, we apply the above lemma to the martingale $\tilde{d}$ from Lemma 30 to obtain the following. Recall that ν was defined at the start of this section as the distribution obtained by applying g to random seeds.

▶ **Lemma 34.** *Given $\tilde{d} : \Sigma^* \to \mathbb{R}_{\geq 0}$, if for all sequences $X \in \mathcal{S}$ where $\nu(\mathcal{S}) > 0$, there exists infinitely many $n \in \mathbb{N}$ such that $\tilde{d}(X \upharpoonright 2^n) > 2^{(1-\tilde{s})2^{n-1}} \tilde{d}(X \upharpoonright 2^{n-1})$, then it holds that for infinitely many n,*

$$\Pr_{x \sim \Sigma^{s_0 \cdot 2^n}}[\tilde{d}(g(x)) > 2^{(1-\tilde{s})2^n} \tilde{d}(g(x \upharpoonright s_0 \cdot 2^{n-1}))] \geq \frac{1}{n^2}. \tag{2}$$

This follows directly from Lemma 33. Indeed, by the hypothesis and the definition of f, every $X \in \mathcal{S}$ satisfies $f(X,n) = 1$ for infinitely many n. Since $\nu(\mathcal{S}) > 0$, Lemma 33 implies that this event has ν-measure at least n^{-c} for infinitely many n, for every $c > 1$. Unwinding the definition of f and using that ν is induced by applying g to uniformly random seeds gives the displayed probability bound; taking $c = 2$ completes the proof.

Now we analyze the performance of the algorithm on outputs of the PRG. Let $n+1$ be one of the lengths at which the condition in Lemma 34 holds. Let $w', w \in \Sigma^{2^n}$ such that $w = G_{2^n}(x)$ for some $x \in \Sigma^{s_0 2^n}$ and $w' = g(r)$ for some $r \in \Sigma^{s_0 2^n}$.

In order for the condition $M(w'w) \geq \mathbf{c} \cdot 2^{(1-\tilde{s})|w|} \cdot M(w')$ to hold, it suffices that $\tilde{d}(w'w) \geq 2^{(1-\tilde{s})|w|} \cdot \tilde{d}(w')$ and $M(w'w) \geq \mathbf{c} \cdot \tilde{d}(w'w)$ and $M(w) \leq \tilde{d}(w)$.

From Lemma 34, it follows that the first condition holds with probability at least $(n+1)^{-2}$ among x, r selected uniformly at random from $\Sigma^{s_0 \cdot 2^n} \times \Sigma^{s_0 \cdot 2^n}$ and taking $w = g(x), w' = g(r)$. Since $\tilde{d}$ is ν-approximable, for some $k \in \mathbb{N}$, on input $w \in \Sigma^{2^n}$, the computations of all $M(w)$ in Algorithm 2 outputs a value in $[\mathbf{c} \cdot \tilde{d}(w), \tilde{d}(w)]$ with probability at least $1 - 2^{-(kn-1)}$ according to the distribution ν_{2^n}. Note that the distribution according to ν_{2^n} corresponds to g applied to the uniform distribution on $2^{s_0 n}$.

Therefore it follows that for infinitely many n,

$$\Pr_{(x,r)\sim\Sigma^{s_0\cdot 2^n}\times\Sigma^{s_0\cdot 2^n}}[A(G_{2^n}(x), r) = 1] \geq \frac{1}{(n+1)^2} - \frac{1}{2^{kn}}.$$

Hence, taking $N = 2^n$, for infinitely many n,

$$\Pr_{x\sim\Sigma^{s_0\cdot N}}[A(G_N(x)) = 1] \geq \frac{1}{(\log N + 1)^2} - \frac{1}{N^k}. \tag{3}$$

4.1.5 Performance on uniformly random inputs

Finally, we show that A outputs 1 with exponentially small probability on a uniformly random input of the same length. This uses the Kolmogorov inequality for supermartingales to bound how many extensions can force a multiplicative jump of the required size, together with the fact that M fails to approximate $\tilde{d}$ only on a ν-small set.

▶ **Lemma 35** (Kolmogorov Inequality [27]). *Let $d : \Sigma^* \to \mathbb{R}_{\geq 0}$ be a supermartingale. For any $w' \in \Sigma^*$ and $n, c \in \mathbb{N}$, the number of strings $w \in \Sigma^n$ such that $d(w'y) \geq c \cdot d(w')$ for some $y \sqsubseteq w$ is less than or equal to $2^n/c$.*

We need an upper bound on the number of strings $w \in \Sigma^{2^n}$ and $r \in \Sigma^{s_0 2^n}$ such that $A(w, r) = 1$. This happens when $M(w'w) \geq \mathbf{c} \cdot 2^{(1-\tilde{s})|w|} \cdot M(w')$.

For any $n \in \mathbb{N}$, take any $r \in \Sigma^{s_0 \cdot 2^n}$, and let $w' = g(r)$. Using the Kolmogorov Inequality for Martingales (Lemma 35), the number of strings $w \in \Sigma^{2^n}$ such that $\tilde{d}(w'w) \geq \mathbf{c}^2 \cdot 2^{|w|(1-\tilde{s})}\tilde{d}(w')$ is less than $2^{|w|}/(\mathbf{c}^2 \cdot 2^{|w|(1-\tilde{s})})$.

For the remaining set of strings, we have that $\tilde{d}(w'w) < \mathbf{c}^2 \cdot 2^{|w|(1-\tilde{s})}\tilde{d}(w')$. If we have that $M(w'w) \leq \tilde{d}(w'w)$ and $M(w') \geq \mathbf{c} \cdot \tilde{d}(w')$, it follows that $A(w, r) = 0$.

We know that, $\nu\{w \in \Sigma^{2^n} : M(w) \notin [\mathbf{c} \cdot \tilde{d}(w), \tilde{d}(w)]\} \leq 2^{-nk}$ for some $k \in \mathbb{N}$. From this, it follows that $\Pr_{r\in\Sigma^{s_0\cdot N}}[M(w') < \mathbf{c} \cdot \tilde{d}(w')] \leq 2^{-nk}$.

Notice that $\{w \in \Sigma^j : M(w) \notin [\mathbf{c} \cdot \tilde{d}(w), \tilde{d}(w)]\} \subseteq \mathrm{supp}(\nu_j)$ for every j. From the block-wise definition of the mapping g, for any fixed $w' \in g(\Sigma^{2^n})$, we obtain that there are at most $2^{s_0 \cdot 2^n}$ strings in the set $\{w'w\}_{w\in\Sigma^{2^n}}$ contained in $\mathrm{supp}(\nu_{2^{n+1}})$. Therefore, for any fixed w', the number of strings $w \in \Sigma^{2^n}$ such that $M(w'w) \notin [\mathbf{c} \cdot \tilde{d}(w'w), \tilde{d}(w'w)]$ is at most $2^{s_0 \cdot 2^n}$. Therefore, we have that

$$\Pr_{x\sim\Sigma^N}\Pr_{r\in\Sigma^{s_0\cdot N}}[A(x, r) = 1] < \mathbf{c}^{-2} \cdot 2^{-N(1-\tilde{s})} + N^{-k} + 2^{-(1-s_0)N}.$$

Therefore, it follows that for any $n \in \mathbb{N}$,

$$\Pr_{x\sim\Sigma^N}[A(x) = 1] < \mathbf{c}^{-c} \cdot 2^{-N(1-\tilde{s})} + N^{-k} + 2^{-(1-s_0)N}.$$

Comparing the two cases yields that for infinitely many lengths, A distinguishes PRG outputs from uniform with non-negligible advantage, contradicting pseudorandomness. Hence $\nu(S^\infty(d')) = 0$ for every polynomial-time ν-approximable s'-supergale with $s' \in (s, 1/2)$.

4.1.6 Proof of Lemma 25

Proof. If one-way functions exist then there exists PRG $G_n : \Sigma^{s_0 n} \to \Sigma^n$ running in time $t_G(n) \in \mathrm{poly}(n)$ [Theorem 18]. Using $\{G_n\}$, we construct a polynomial-time samplable distribution ν over Σ^∞ as above. By Corollary 29 and the choice of s_0, we have $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \leq 2s_0 < s$ for almost every $X \sim \nu$.

From the arguments in Sections 4.1.4 and 4.1.5, we obtain the following consequence. Suppose there exists some $s' \in (s, 1/2)$ and a polynomial-time ν-approximable s'-supergale d such that $\nu(S^\infty(d)) > 0$. Then there exists a distinguisher algorithm A running in time $t_A(n) \in \mathrm{poly}(n)$ such that, for infinitely many $N \in \mathbb{N}$ and for all $k \in \mathbb{N}$,

$$\Pr_{x \in \Sigma^{s_0 \cdot N}}[A(G_N(x)) = 1] - \Pr_{x \in \Sigma^N}[A(x) = 1] > \frac{1}{(\log N + 1)^2} - \frac{2}{N^k} - \frac{1}{\mathbf{c}^2 \cdot 2^{N(1-\tilde{s})}} - \frac{1}{2^{N(1-s_0)}}.$$

Let c be any constant such that $t_A(n) \leq n^c$ and for all large enough N,

$$\frac{1}{(\log N + 1)^2} - \frac{2}{N^k} - \frac{1}{\mathbf{c}^2 \cdot 2^{N(1-\tilde{s})}} - \frac{1}{2^{N(1-s_0)}} \geq \frac{1}{N^c}.$$

It follows that G_N is not a $(N^c, 1/N^c)$ PRG for infinitely many N. Therefore the family $\{G_n\}_{n \in \mathbb{N}}$ is not a PRG, contradicting the assumption that one-way functions exist. This completes the proof of the lemma. ◀

4.2 Infinitely-often one-way functions from dimension gaps

We now explain the reverse direction: an almost-everywhere dimension gap over an efficiently samplable source can be converted into infinitely-often one-way functions.

▶ **Lemma 36.** *If there exist an $s < 1$ and a polynomial-time samplable distribution ν over Σ^∞ such that:*

1. *$\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \leq s$ for ν-almost every X.*
2. *There exists $\hat{s} \in (s, 1]$ such that for every $s' \in (s, \hat{s})$ and every polynomial-time ν-approximable s'-supergale d, we have $\nu(S^\infty(d)) = 0$.*

Then infinitely-often one-way functions exist.

Proof. Assume the hypothesis, and toward a contradiction assume that infinitely-often one-way functions do not exist. Fix reals $s < s' < s'' < s''' < \hat{s}$ with $2^{s'''} \in \mathbb{Q}$, and define

$$d(w) := 2^{s'''|w|}\, \nu(w).$$

It is straightforward that d is an s'''-supergale. The low-$\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ condition implies that with ν-probability close to 1, draws $X \sim \nu$ have many prefixes $X \upharpoonright n$ with small time-bounded description length (say $\leq s'n$ for all large n). For each n, there are at most $2^{s'n}$ such "typical" strings of length n. If all these typical strings had ν_n-mass $< 2^{-s''n}$ for all large n, then the total ν_n-mass of typical length-n strings would be at most $2^{s'n} \cdot 2^{-s''n} = 2^{-(s''-s')n}$, contradicting that typical strings carry almost all the mass. Therefore, with *positive* ν-probability, infinitely many n satisfy $\nu_n(X \upharpoonright n) \geq 2^{-s''n}$, and along those n we have $d(X \upharpoonright n) \geq 2^{(s'''-s'')n}$; hence $\nu(S^\infty(d)) > 0$.

We now formalize the positive-measure conclusion used above.

▶ **Lemma 37.** $\nu\{X \in \Sigma^\infty : \exists^\infty n \text{ such that } \nu_n(X \upharpoonright n) \geq 2^{-n \cdot s''}\} > 0$.

Notice that if $\nu_n(w) \geq 2^{-n \cdot s''}$ then $d(w) \geq 2^{n(s'''-s'')}$. Therefore from Lemma 37 we obtain that $\nu\{X \in \Sigma^\infty : \limsup_{n\to\infty} d(X \upharpoonright n) = \infty\} > 0$.

On the other hand, under the assumption that infinitely-often one-way functions do not exist, we invoke a distributional estimation theorem for polynomial-time samplable distributions. Distributional estimation results of this kind go back to Impagliazzo and Luby [20] and Impagliazzo and Levin [19]. Ilango, Ren, and Santhanam [18] give an algorithm that, for every polynomial-time samplable distribution ensemble $\mathcal{D} = \{\mathcal{D}_n\}$, estimates $\mathcal{D}_n(w)$ within a constant multiplicative factor on all but a $\mathcal{D}_n$-small exceptional set of samples $w \sim \mathcal{D}_n$. Nanashima and Hirahara [13] prove a substantially stronger form, giving a multiplicative $(1 \pm \delta)$ approximation (for arbitrarily small $\delta > 0$) on almost all samples from $\mathcal{D}_n$. Such distributional estimation results can be used to obtain efficient algorithms for universal extrapolation and Solomonoff's inductive inference. For our purposes in this paper, the constant-factor guarantee from [18] is sufficient, and we apply it to the particular samplable distribution $\nu_n(w) = \nu(C_w)$. Combining this estimator with a standard gale normalization step yields a polynomial-time ν-approximator for $\nu(C_w)$ on $\mathrm{supp}(\nu_n)$, and therefore for $d(w) = 2^{s'''|w|}\nu(C_w)$ (using $2^{s'''} \in \mathbb{Q}$). Thus d is a polynomial-time ν-approximable s'''-supergale with $\nu(S^\infty(d)) > 0$, contradicting item (2). Hence infinitely-often one-way functions must exist. The formal estimation result and the approximability argument are given next.

▶ **Theorem 38** ([18, 20, 19, 13])**.** *Assume that no infinitely-often one-way functions exist. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be a polynomial-time samplable distribution and $q \geq 1$ be an arbitrary constant. Then, there exists a probabilistic polynomial-time algorithm $\mathcal{A}$ and constant $c < 1$ such that for all n,*

$$\Pr_{x\sim\mathcal{D}_n}[c \cdot \mathcal{D}_n(x) \leq \mathcal{A}(x) \leq \mathcal{D}_n(x)] \geq 1 - O\left(\frac{1}{n^q}\right).$$

▷ Claim 39. d is polynomial-time ν-approximable.

Proof. Let S be the machine that samples the distribution ν such that for every n, $\Pr_r[S(1^n, r) = w] = \nu_n(w)$ for every $w \in \Sigma^n$. Let c' be the constant such that S uses $n^{c'}$ random bits to sample a string of length n.

Assume that no infinitely-often one-way functions exist. Then, the function $f : \Sigma^* \to \Sigma^*$ defined as

$$f(x) = S(1^{|x|^{1/c'}}, x)$$

is not an infinitely-often weak one-way function. Hence for any $q \geq 1$ there exist an algorithm $\mathcal{I}$ such that for all n,

$$\Pr_{r\sim\Sigma^{n^{c'}}}[f(\mathcal{I}(f(r))) = f(r)] = \Pr_{w\sim\nu_n}[f(\mathcal{I}(w)) = w] \geq 1 - O\left(n^{-q}\right). \tag{4}$$

From our assumption that infinitely-often one-way functions do not exist and Theorem 38, we obtain that for any $q \geq 1$, there exists a probabilistic polynomial-time algorithm $\mathcal{A}$ and constant $c < 1$ such that for all n,

$$\Pr_{w\sim\nu_n}[c \cdot \nu_n(w) \leq \mathcal{A}(w) \leq \nu_n(w)] \geq 1 - O\left(n^{-q}\right). \tag{5}$$

Finally, we give the algorithm $\mathcal{M}$ that ν-approximates d. $\mathcal{M}$ on input $w \in \Sigma^n$ runs $\mathcal{I}$ on w. If $f(\mathcal{I}(w)) \neq w$, $\mathcal{M}$ outputs 0. Else, $\mathcal{M}$ runs $\mathcal{A}$ on w and outputs $2^{s'''|w|}\mathcal{A}(w)$. Since $2^{s'''} \in \mathbb{Q}$, $\mathcal{M}$ runs in polynomial time. For any w such that $\nu(w) = 0$ (i.e. $w \notin \mathrm{supp}(\nu_n)$), $\mathcal{I}$

will never succeed in finding a pre-image r such that $f(r) = w$ and hence $\mathcal{M}$ always outputs 0. Therefore for every n, $\{w : \mathcal{M}(w) \notin [c \cdot d(w), d(w)]\} \subseteq \text{supp}(\nu_n)$. Now, from (4) and (5) we obtain that,

$$\Pr_{x\sim\nu_n}[c \cdot d(x) \leq \mathcal{M}(x) \leq d(x)] \geq 1 - O\left(\frac{1}{n^q}\right) - O\left(\frac{1}{n^q}\right) \geq 1 - O\left(\frac{1}{n^q}\right).$$

Therefore for every n, $\nu\{w : \mathcal{M}(w) \notin [c \cdot d(w), d(w)]\} \leq O\left(n^{-q}\right)$. Since $\mathcal{A}$ runs in randomized polynomial time and q is arbitrary, $\mathcal{M}$ witnesses the fact that d is a polynomial-time ν-approximable s'''-gale. ◁

The claim shows that d is a polynomial-time ν-approximable s'''-supergale. Since $s''' < \hat{s}$ and $\nu(S^\infty(d)) > 0$, this contradicts the second item in the hypothesis. Therefore, infinitely-often one-way functions exist. ◀

4.3 Dimension gaps over sequences from one-way functions

Lemma 25 is a measure statement: under one-way functions it gives a samplable distribution ν on Σ^∞ for which typical sequences have low $\mathcal{K}^{\text{str}}_{\text{poly}}$, but every polynomial-time ν-approximable gale of exponent $< 1/2$ succeeds only on a ν-null set. We now indicate how this measure-theoretic gap yields an *individual* sequence with $\mathcal{K}^{\text{str}}_{\text{poly}}$ small but $\dim_{\text{P}}$ bounded away from it, and then how the same strategy can be pushed to obtain gaps arbitrarily close to 1.

▶ **Theorem 22.** *If for all $X \in \Sigma^\infty$, $\dim_{\text{P}}(X) = \mathcal{K}_{\text{poly}}(X)$, then no one-way functions exist.*

We first record the gale-combination construction needed to pass from pointwise polynomial-time gales, whose running-time exponents may vary with the sequence, to a single polynomial-time gale on a positive-measure set.

▶ **Lemma 40.** *For any time $t(n) \in \text{poly}(n)$, and $s > 0$ with $2^s \in \mathbb{Q}$, there exists an exact $t(n) \cdot n \cdot \log(n)$-time computable s-gale $\tilde{d}$ such that for all exact $t(n)$-time computable s-gales d, there exist a constant c such that for all $X \in \Sigma^\infty$ and $n \in \mathbb{N}$, $\tilde{d}(X \upharpoonright n) \geq c \cdot d(X \upharpoonright n)$.*

We now prove a stronger theorem. Theorem 22 follows from Theorem 41, since for all sequences $X \in \Sigma^\infty$, $\mathcal{K}_{\text{poly}}(X) \leq \mathcal{K}^{\text{str}}_{\text{poly}}(X)$.

▶ **Theorem 41.** *If one-way functions exist, then for every $s < 1/2$, there exist $X \in \Sigma^\infty$ such that $\mathcal{K}^{\text{str}}_{\text{poly}}(X) \leq s$ and $\dim_{\text{P}}(X) \geq 1/2$.*

Proof. Assume that one-way functions exist. From the proof of Lemma 25, we obtain that for every $s < 1/2$ there exist a polynomial-time samplable distribution ν over Σ^∞ and polynomial t such that:

1. The number of random bits used by the sampler for ν on input 1^n is at most sn.
2. For every $s' \in (s, 1/2)$ and any polynomial-time ν-approximable s'-supergale d, $\nu(S^\infty(d)) = 0$.

Consider the canonical closed full-measure support set $\mathcal{F} = \bigcap_{n\geq 1} \bigcup_{w\in\text{supp}(\nu_n)} C_w$. Now, $\cup_{w\in\text{supp}(\nu_n)} C_w$ is a finite union of cylinder sets in Σ^∞ and hence is a closed set. Since ν is a polynomial-time samplable probability distribution on Σ^∞, $\cup_{w\in\text{supp}(\nu_n)} C_w$ is a superset of $\cup_{w\in\text{supp}(\nu_{n+1})} C_w$ for every n. Since $\mathcal{F}$ is an intersection of a decreasing sequence of non-empty closed sets in the compact space Σ^∞, $\mathcal{F}$ is non-empty (see [35]). Since $\nu(\cup_{w\in\text{supp}(\nu_n)} C_w) = 1$ for every n, using the continuity of probability measures from above [2], we get $\nu(\mathcal{F}) = 1$. Since the number of random bits used by the sampler for ν on input 1^n is at most sn, $\mathcal{K}^{\text{str}}_{\text{poly}}(X) \leq s$ for every $X \in \mathcal{F}$.

For the sake of contradiction, assume that there exists $s' \in (s, 1/2)$ such that $\dim_{\mathrm{P}}(X) < s'$ for every $X \in \mathcal{F}$. So, for every $X \in \mathcal{F}$ there exists an exact computable s'-gale d'_X that runs in time $t_{d'_X}(n) \in \mathrm{poly}(n)$, such that d'_X succeeds on X. We group $\mathcal{F}$ according to a polynomial running-time exponent. Define

$$R_i = \left\{ X \in \mathcal{F} \;\middle|\; \text{there exists an } n^i\text{-time } s'\text{-gale } d \text{ such that } X \in S^\infty(d) \right\},$$

and let $S_i = R_i \setminus R_{i-1}$. Note that $\{S_i\}_{i\in\mathbb{N}}$ is a partition of $\mathcal{F}$ into countably many disjoint subsets. As $\sum_{i=1}^{\infty} \nu(S_i) = 1$, there exists an $k \in \mathbb{N}$ such that $\sum_{i=1}^{k} \nu(S_i) > 0$. Define $\mathcal{S} = \bigcup_{i=1}^{k} S_i$. On this positive-measure subset, all witnessing gales run within a common polynomial time bound. Using Lemma 40, there exists an exact polynomial-time computable s'-gale d that succeeds on all sequences in $\mathcal{S}$. Since d is polynomial-time computable, it is automatically polynomial-time ν-approximable. Now, since we know that $\nu(S^\infty(d)) \geq \nu(\mathcal{S}) > 0$, we obtain a contradiction to Lemma 25. Hence, it must be the case that there exists a sequence $X \in \mathcal{F}$ with $\dim_{\mathrm{P}}(X) > s'$. Since s' was arbitrary, the theorem follows. ◀

The same argument also gives an almost-sure version over the samplable source obtained from Lemma 25.

▶ **Lemma 42.** *If one-way functions exist then for some $s < 1/2$, there exist a polynomial-time samplable distribution ν over Σ^∞ such that:*

1. *The number of random bits used by the sampler for ν on input 1^n is at most sn.*
2. *For every $s' \in (s, 1/2)$ and polynomial-time ν-approximable s'-supergale d, $\nu(S^\infty(d)) = 0$.*
3. *$\nu\{X \in \Sigma^\infty : \mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \leq s$ and $\dim_{\mathrm{P}}(X) \geq 1/2 > s\} = 1$.*

We now strengthen the conclusion by amplifying the gap. The above route naturally yields the threshold $1/2$ because the forward implication only rules out success of *ν-approximable* gales below $1/2$. To push the separation to $1 - \varepsilon$, we keep the same blueprint (construct a low-$\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ source and diagonalize against efficient betting strategies), but we modify the diagonalization so that it is carried out against *exact* computable gales. This avoids the accumulation of approximation slack, which becomes the bottleneck near dimension 1, and allows us to push the lower bound on $\dim_{\mathrm{P}}$ to $1 - \varepsilon$ while keeping $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$ below ε.

▶ **Theorem 23.** *If one-way functions exist, then for any $\epsilon > 0$, there exists $X \in \Sigma^\infty$ such that $\dim_{\mathrm{P}}(X) - \mathcal{K}_{\mathrm{poly}}(X) \geq 1 - \epsilon$.*

We prove the following stronger statement independently since it does not follow directly using the techniques in the proofs of Lemma 25 and Theorem 41. The main obstacle is that techniques involving approximable gales introduce non-negligible approximation errors while trying to achieve gaps close to 1. In the proof of the below theorem we instead employ computable gales. Whether Lemma 25 can be strengthened to yield such a gap remains an open question.

▶ **Theorem 43.** *If one-way functions exist, then for every $\epsilon > 0$ there exist set $\mathcal{F}$ and a sequence X such that $\dim_{\mathrm{P}}(\mathcal{F}) - \mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(\mathcal{F}) \geq 1 - \epsilon$ and $\dim_{\mathrm{P}}(X) - \mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) \geq 1 - \epsilon$ respectively.*

The argument follows the same overall blueprint as the proof of Theorem 22: we use the PRG-induced mapping g to produce sequences with low $\mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}$, and then diagonalize against efficient betting strategies by converting a successful s'-gale into a martingale that must exhibit large multiplicative gains at dyadic checkpoints, which yields a distinguisher for the underlying PRG. The only substantive change is that, to obtain gaps arbitrarily close to

1, we carry out the diagonalization using exact polynomial-time computable gales rather than ν-approximable gales, so that no approximation slack accumulates when we test many intermediate prefixes. A similar argument yields the corresponding separation statement for sets, by applying the same support/compactness reduction and the same diagonalization idea at the level of collections rather than individual sequences.

We obtain the following corollary of Theorem 43. This result provides a strong negative answer to the open question posed by Stull in [36] under the assumption that one-way functions exist.

▶ **Corollary 44.** *If one-way functions exist then for every $\epsilon < 1/2$ there exists $X \in \Sigma^\infty$ such that $\mathcal{K}_{\mathrm{poly}}(X) \leq \mathcal{K}^{\mathrm{str}}_{\mathrm{poly}}(X) < \epsilon < 1 - \epsilon < \dim_{\mathrm{P}}(X) \leq \mathrm{Dim}_{\mathrm{P}}(X)$.*

5 Open Problems and Discussion

Our findings suggest several promising directions for future research. A key open question is whether weaker hardness assumptions, such as P $\neq$ NP, DistNP $\not\subseteq$ AvgP or DistNP $\not\subseteq$ AvgBPP, can be characterized in terms of separations of polynomial-time dimension, as defined using time-bounded s-gales and time-bounded Kolmogorov complexity in appropriate settings. Related work on time-bounded Kolmogorov complexity has also revealed connections with average-case hardness and one-way functions [25, 9]. Specifically, can these hardness assumptions be equivalently expressed through if-and-only-if statements involving polynomial-time dimension?

A different direction is to seek finite-length versions of these phenomena. Polynomial-time dimension is asymptotic, while the pseudorandomness assumptions used in this paper are finite-length indistinguishability assumptions. It would be interesting to define finite-length measures based on efficient online betting strategies, or equivalently log-loss prediction on n-bit samples, and compare them with finite-time Kolmogorov complexity rates. Can one obtain finite-length separation theorems between efficient betting and efficient compression that characterize one-way functions, average-case hardness, or other standard assumptions?

Another natural direction concerns extraction and dimension amplification. Standard randomness extractors transform weak finite sources into distributions close to uniform [37], while dimension measures asymptotic randomness rates along individual infinite sequences. Can seeded extractors, condensers, or related pseudorandomness objects be lifted to transformations that increase polynomial-time dimension for typical outputs of efficiently samplable infinite sources? A complementary question is whether cryptographic hardness can be used to construct efficiently samplable sources whose finite prefixes have substantial entropy, but whose polynomial-time dimension cannot be increased by these transformations.

Our results reveal a duality between the non-robustness of polynomial-time dimension and the existence of one-way functions. An interesting open question is whether notions of polynomial-time randomness, defined via betting algorithms and Kolmogorov incompressibility of prefixes (see [36] for more details), capture equivalent concepts of randomness. Polynomial-time betting algorithms that succeed on specific sequences do not necessarily provide guarantees on the rate of capital accumulation in general. Consequently, investigating the robustness of polynomial-time randomness and its potential connections to cryptographic and complexity-theoretic primitives may require fundamentally new approaches.

References

1 Krishna B Athreya, John M Hitchcock, Jack H Lutz, and Elvira Mayordomo. Effective strong dimension in algorithmic information and computational complexity. *SIAM journal on computing*, 37(3):671–705, 2007. doi:10.1137/S0097539703446912.

2 Patrick Billingsley. *Probability and measure*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, Inc., New York, third edition, 1995. A Wiley-Interscience Publication.

3 Andrej Bogdanov and Luca Trevisan. Average-case complexity. *Found. Trends Theor. Comput. Sci.*, 2(1):1–106, December 2006. doi:10.1561/0400000004.

4 Chris Bourke, John M. Hitchcock, and N. V. Vinodchandran. Entropy rates and finite-state dimension. *Theoretical Computer Science*, 349(3):392–406, 2005. doi:10.1016/J.TCS.2005.09.040.

5 Jack J. Dai, James I. Lathrop, Jack H. Lutz, and Elvira Mayordomo. Finite-state dimension. *Theoretical Computer Science*, 310(1-3):1–33, 2004. doi:10.1016/S0304-3975(03)00244-5.

6 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.

7 Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic randomness and complexity*. Theory and Applications of Computability. Springer, New York, 2010. doi:10.1007/978-0-387-68441-3.

8 Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In Harriet Ortiz, editor, *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 416–426. ACM, 1990. doi:10.1145/100216.100272.

9 Halley Goldberg and Valentine Kabanets. Consequences of Randomized Reductions from SAT to Time-Bounded Kolmogorov Complexity. In Amit Kumar and Noga Ron-Zewi, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024)*, volume 317 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 51:1–51:19, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.APPROX/RANDOM.2024.51.

10 Oded Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2001.

11 Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, 2008. doi:10.1017/CBO9780511804106.

12 Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984. doi:10.1016/0022-0000(84)90070-9.

13 Shuichi Hirahara and Mikito Nanashima. Learning in pessiland via inductive inference. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 447–457. IEEE, 2023. doi:10.1109/FOCS57990.2023.00033.

14 J.M. Hitchcock and N.V. Vinodchandran. Dimension, entropy rates, and compression. In *Proceedings. 19th IEEE Annual Conference on Computational Complexity, 2004.*, pages 174–183, 2004. doi:10.1109/CCC.2004.1313835.

15 John M. Hitchcock. Effective fractal dimension: Foundations and applications. Ph.D. thesis, Iowa State University, 2003.

16 John M. Hitchcock and N.V. Vinodchandran. Dimension, entropy rates, and compression. *Journal of Computer and System Sciences*, 72(4):760–782, 2006. doi:10.1016/j.jcss.2005.10.002.

17 Johan HÅstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. doi:10.1137/S0097539793244708.

18 Rahul Ilango, Hanlin Ren, and Rahul Santhanam. Robustness of average-case meta-complexity via pseudorandomness. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1575–1583, 2022. doi:10.1145/3519935.3520051.

19 R. Impagliazzo and L. A. Levin. No better ways to generate hard np instances than picking uniformly at random. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 812–821, 1990. doi:10.1109/FSCS.1990.89604.

20 R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235, 1989. doi:10.1109/SFCS.1989.63483.

21 Valentine Kabanets and Jin-Yi Cai. Circuit minimization problem. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 73–79, New York, NY, USA, 2000. Association for Computing Machinery. doi:10.1145/335305.335314.

22 Andrei N Kolmogorov. Three approaches to the quantitative definition ofinformation'. *Problems of information transmission*, 1(1):1–7, 1965.

23 L. A. Levin. The tale of one-way functions. *Probl. Inf. Transm.*, 39(1):92–103, January 2003. doi:10.1023/A:1023634616182.

24 Ming Li and Paul Vitányi. *An introduction to Kolmogorov complexity and its applications*, volume 3. Springer, 2008.

25 Yanyi Liu and Rafael Pass. On one-way functions and kolmogorov complexity. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1243–1254, 2020. doi:10.1109/FOCS46700.2020.00118.

26 María López-Valdés and Elvira Mayordomo. Dimension is compression. In Joanna Jędrzejowicz and Andrzej Szepietowski, editors, *Mathematical Foundations of Computer Science 2005*, volume 3618 of *Lecture Notes in Computer Science*, pages 676–685, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. doi:10.1007/11549345_58.

27 Jack H. Lutz. Dimension in complexity classes. *SIAM J. Comput.*, 32(5):1236–1259, 2003. doi:10.1137/S0097539701417723.

28 Jack H. Lutz. The dimensions of individual strings and sequences. *Information and Computation*, 187(1):49–79, 2003. doi:10.1016/S0890-5401(03)00187-1.

29 Elvira Mayordomo. A Kolmogorov complexity characterization of constructive Hausdorff dimension. *Inform. Process. Lett.*, 84(1):1–3, 2002. doi:10.1016/S0020-0190(02)00343-5.

30 Moni Naor. Bit commitment using pseudorandomness. *J. Cryptol.*, 4(2):151–158, January 1991. doi:10.1007/BF00196774.

31 Andre Nies. *Computability and Randomness*. Oxford University Press, Inc., USA, 2009.

32 K.W. Regan and D. Sivakumar. Probabilistic martingales and bptime classes. In *Proceedings. Thirteenth Annual IEEE Conference on Computational Complexity (Formerly: Structure in Complexity Theory Conference) (Cat. No.98CB36247)*, pages 186–200, 1998. doi:10.1109/CCC.1998.694604.

33 John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Symposium on the Theory of Computing*, 1990. URL: https://api.semanticscholar.org/CorpusID:10858937.

34 Halsey Lawrence Royden and Patrick Fitzpatrick. *Real analysis*, volume 32. Macmillan New York, 1988.

35 George F. Simmons. *Introduction to Topology and Modern Analysis*. McGraw-Hill, New York, 1963.

36 Donald M. Stull. Resource bounded randomness and its applications. In Johanna N. Y. Franklin and Christopher P. Porter, editors, *Algorithmic Randomness: Progress and Prospects*, Lecture Notes in Logic, pages 301–348. Cambridge University Press, Cambridge, 2020.

37 Salil P. Vadhan. Pseudorandomness. *Foundations and Trends® in Theoretical Computer Science*, 7(1–3):1–336, 2012. doi:10.1561/0400000010.

38 Andrew C. Yao. Protocols for secure computations. In *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*, pages 160–164, 1982. doi:10.1109/SFCS.1982.38.