The Space Complexity of Sampling

Eshan Chattopadhyay
Cornell University, Ithaca, NY, USA

Jesse Goodman
Cornell University, Ithaca, NY, USA

David Zuckerman
University of Texas at Austin, TX, USA

Abstract

Recently, there has been exciting progress in understanding the complexity of distributions. Here, the goal is to quantify the resources required to generate (or sample) a distribution. Proving lower bounds in this new setting is more challenging than in the classical setting, and has yielded interesting new techniques and surprising applications. In this work, we initiate a study of the complexity of sampling with *limited memory*, and obtain the first nontrivial sampling lower bounds against oblivious read-once branching programs (ROBPs).

In our first main result, we show that any distribution sampled by an ROBP of width $2^{\Omega(n)}$ has statistical distance $1 - 2^{-\Omega(n)}$ from any distribution that is uniform over a good code. More generally, we obtain sampling lower bounds for any list decodable code, which are nearly tight. Previously, such a result was only known for sampling in AC^0 (Lovett and Viola, CCC'11; Beck, Impagliazzo and Lovett, FOCS'12). As an application of our result, a known connection implies new data structure lower bounds for storing codewords.

In our second main result, we prove a direct product theorem for sampling with ROBPs. Previously, no direct product theorems were known for the task of sampling, for any computational model. A key ingredient in our proof is a simple new lemma about amplifying statistical distance between sequences of somewhat-dependent random variables. Using this lemma, we also obtain a simple new proof of a known lower bound for sampling disjoint sets using two-party communication protocols (Göös and Watson, RANDOM'19).

2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography

Keywords and phrases Complexity of distributions, complexity of sampling, extractors, list decodable codes, lower bounds, read-once branching programs, small-space computation

Digital Object Identifier 10.4230/LIPIcs.ITCS.2022.40

Related Version *Full Version*: `https://eccc.weizmann.ac.il/report/2021/106/`

Funding *Eshan Chattopadhyay*: Research supported by NSF CAREER Award 2045576.
Jesse Goodman: Research supported by NSF CAREER Award 2045576.
David Zuckerman: Research supported by NSF Grants CCF-1705028 and CCF-2008076 and a Simons Investigator Award (#409864).

Acknowledgements We thank William Hoza and anonymous reviewers for extremely helpful comments.

1 Introduction

A central goal in complexity theory is to quantify the resources required to perform certain tasks. Traditionally, complexity theory has focused on the task of *computing*: here, one fixes a function $f : \{0,1\}^m \to \{0,1\}^n$ and computational model $\mathcal{C}$ (e.g., low-depth circuits), and asks for lower bounds on the size of any $F \in \mathcal{C}$ that computes f.

Recently, a growing body of work has sought to understand the power of these same computational models for the task of *sampling*. Here, instead of fixing a function f, one picks a target distribution $\mathbf{Q} \sim \{0,1\}^n$. Then, one asks for lower bounds on the size of any $F \in \mathcal{C}$ that generates (samples) $\mathbf{Q}$, when supplied with uniformly random bits.

Following the earlier works of Ambainis, Schulman, Ta-Shma, Vazirani and Wigderson [2] and Goldreich, Goldwasser, and Nussboim [13], Viola was the first to launch a systematic study on the *complexity of sampling distributions* [22]. Since then, this new area of complexity theory has seen an exciting wave of interest [23, 21, 5, 24, 12, 17, 1, 27, 6, 28, 25, 29, 26, 14]. Despite this significant progress, results are still only known for a few computational models like AC^0 and communication protocols. In particular, little is known about the complexity of sampling with *limited memory*, while this remains a fundamental model in other areas of complexity.

In this work, we aim to fill this gap, and initiate a study of the *space* complexity of sampling. Our model will correspond to the streaming model of computation, an active area of research. Our work makes progress on the research program initiated by Viola, who has advocated for the pursuit of sampling lower bounds against every model for which we already have classical lower bounds (including branching programs, Turing machines, and polynomials) [24].

1.1 Key questions

Before we formally introduce our model and present our results, we briefly survey sampling in AC^0, and motivate some key questions about sampling with limited memory.

Sampling can be easier than computing. A motivating paradigm in the complexity of sampling is the (perhaps surprising) fact that a fixed computational model $\mathcal{C}$ may be more powerful at sampling than computing. In particular, consider fixing a function $f : \{0,1\}^n \to \{0,1\}^m$ and comparing the tasks of computing f on every input x, with sampling $f(\mathbf{U}_n)$ [13]. Intuitively, the latter task should seem easier: any $F \in \mathcal{C}$ that computes f must also have $F(\mathbf{U}_n) = f(\mathbf{U}_n)$. Furthermore, setting f^{-1} to be a one-way permutation makes $f(x)$ very hard to compute, but $f(\mathbf{U}_n)$ very easy to sample [22].

Amazingly, we also have examples of extremely simple *explicit* functions that demonstrate this separation. The canonical example is the function $f(x) = (x, \mathsf{parity}(x))$: the celebrated result of Håstad [15] shows that f cannot be computed in AC^0, yet Babai [3], Boppana and Lagarias [7] give an extremely simple AC^0 circuit that samples $(\mathbf{U}_n, \mathsf{parity}(\mathbf{U}_n))$. Thus, obtaining sampling lower bounds is strictly more challenging (at least in AC^0), and their pursuit may unveil exciting new techniques and applications [22].

A natural first question, then, is to ask whether this motivation still holds in the limited memory setting:

▶ **Question 1.** *Does there exist an explicit boolean function $b : \{0,1\}^n \to \{0,1\}$ that is hard to compute with limited memory, but such that $(\mathbf{U}_n, b(\mathbf{U}_n))$ is easy to sample with limited memory?*

Sampling lower bounds for input-output pairs. Above, we saw that for the parity function $b : \{0,1\}^n \to \{0,1\}$, it holds that $(x, b(x))$ is hard to compute in AC^0, yet $(\mathbf{U}_n, b(\mathbf{U}_n))$ is easy to sample in AC^0. Given this observation, Viola raised the challenge [22] of finding a distribution of the form $(\mathbf{U}_n, b(\mathbf{U}_n))$ that is *hard* to sample in AC^0. In a recent paper [26], Viola provided a strong solution to this challenge, by giving an explicit function $b : \{0,1\}^n \to \{0,1\}$ such that for any AC^0 circuit $F : \{0,1\}^\ell \to \{0,1\}^{n+1}$, it holds that $|F(\mathbf{U}_\ell) - (\mathbf{U}_n, b(\mathbf{U}_n))| \geq \frac{1}{2} - 2^{-n^{\Omega(1)}}$, where $|\cdot|$ denotes statistical distance.

Assuming Question 1 can be answered positively, it is natural to ask whether a similar result holds for sampling in the limited memory setting:

▶ **Question 2.** *Does there exist an explicit boolean function* $b : \{0,1\}^n \to \{0,1\}$ *such that* $(\mathbf{U}_n, b(\mathbf{U}_n))$ *is hard to sample with limited memory?*

Sampling lower bounds for codes. It is easy to see sampling lower bounds for distributions of the form $(\mathbf{U}_n, b(\mathbf{U}_n))$ cannot exceed $1/2$ (since $(\mathbf{U}_n, 0)$ or $(\mathbf{U}_n, 1)$ will yield an upper bound of $1/2$, and both of these are trivial to sample). A complementary question, suggested by Viola [22], is to find other natural distributions $\mathbf{Q} \sim \{0,1\}^n$ with much stronger sampling lower bounds - perhaps even approaching 1.

In 2012, Viola and Lovett demonstrated a distribution of exactly this type [21], setting $\mathbf{Q} \sim \{0,1\}^n$ to be uniform over an asymptotically good error-correcting code, i.e., one having constant relative distance and rate. They showed that for such a distribution $\mathbf{Q}$, any AC^0 circuit $F : \{0,1\}^\ell \to \{0,1\}^n$ has $|F(\mathbf{U}_\ell) - \mathbf{Q}| \geq 1 - \varepsilon$ for $\varepsilon = n^{-\Omega(1)}$. In a subsequent work [5], Beck, Impagliazzo, and Lovett improved the statistical distance to $1 - \varepsilon$ for $\varepsilon = 2^{-n^{\Omega(1)}}$. Using an observation of Viola [22], both works also obtain data structure lower bounds for storing codewords. Given these results, we would like to know:

▶ **Question 3.** *Are good codes hard to sample with limited memory?*

Direct product theorems. Thus far, our questions have asked for small-space analogs of key results known for the complexity of sampling in AC^0. For our final question, we ask for a type of result that has yet to be studied in the complexity of sampling, but which has been well-explored within classical complexity. In particular, we ask for a *direct product theorem*.

In classical complexity, direct product theorems (e.g., Yao's XOR Lemma [30]) are used for hardness amplification: such a result roughly says that if a function f is somewhat hard to compute for a given computational model, then t independent copies of f are *very* hard to compute for that same model. Direct product theorems offer a concrete way to (i) construct simple functions with strong (average-case) lower bounds, and thus (ii) establish strong (average-case) complexity separations between complexity classes.

It is natural to ask whether such direct product theorems can also be established for the task of sampling. In the context of sampling, a direct product theorem can be defined as a result which asserts the following: if a distribution $\mathbf{Q} \sim \{0,1\}^n$ has statistical distance δ from any distribution sampled by some computational model, then t independent copies of $\mathbf{Q}$ (concatenated together) has statistical distance $\gg \delta$ from any distribution sampled by that same model. Our final question is as follows.

▶ **Question 4.** *Can a direct product theorem be established for distributions sampled in limited memory?*

In this paper, we make progress on these four questions. Before presenting our results, we must discuss our model for sampling distributions with limited memory.

1.2 Sampling in small space using oblivious ROBPs

To model sampling with limited memory, we will use the classic model of *oblivious read-once branching programs* (ROBPs). This model corresponds to the streaming model of computation, and thus a better understanding of the power of ROBPs for sampling tasks may also help provide new insights and tools for streaming algorithms.

A first attempt to model sampling in limited memory uses the classic definition of an ROBP (Definition 9), and replaces its input with uniform bits. However, such an ROBP computes a function $f : \{0,1\}^\ell \to \{0,1\}$, and so the distribution $f(\mathbf{U}_\ell)$ it samples will be over $\{0,1\}$. To sample general distributions $\mathbf{Q} \sim \{0,1\}^n$, we need an ROBP that can output multiple bits.

Perhaps the most natural way to extend an ROBP to output multiple bits is to simply allow it to output a sequence of bits upon reading any input bit. More formally, we can assign each edge in the ROBP an additional label consisting of a string of output bits. Then, given an input $x \in \{0,1\}^\ell$, the ROBP traverses a path in the usual way, but now outputs all the output labels seen along the way. Indeed, this is exactly a "read-once" version of multi-output branching programs considered in previous works [9, 8, 4].

It will also be convenient to make one simplifying assumption: just as the inputs in an ROBP are "layered", we will assume that the outputs are also layered. That is, we require that any two edges traversing between the same two layers are labeled with the same number of output bits. This is a natural way to guarantee that the ROBP will compute a function of the form $F : \{0,1\}^\ell \to \{0,1\}^n$, since all paths are guaranteed to output the same number of bits. This completes our definition of *multi-output ROBP* (see Definition 10 for a more formal definition).

Just as a standard ROBP models an algorithm that reads from an input stream, multi-output ROBPs also allow the algorithm to *write* to an output stream (since it may write an arbitrary number of bits at each time step, without storing any of them in its memory). As it turns out, we will also prove that sampling using this model is equivalent (up to a small loss in parameters) to sampling using a different natural model from the field of randomness extractors [18]. Furthermore, note that for functions with one bit of output, our definition is equivalent to the classic single-bit-output ROBP definition. Thus, we will henceforth refer to multi-output ROBPs simply as ROBPs.

1.3 Summary of our main results

With our questions in mind and our model formally defined, we are ready to state our results. Qualitatively, we provide positive answers to all four questions from Section 1.1.

Question 1 and Question 2 are straightforward to answer by applying known (or easy-to-prove) lower bounds from communication complexity. We discuss these results in Section 4. On the other hand, Question 3 and Question 4 are more challenging to resolve, and our two main contributions are positive answers to these questions. We go into more detail below.

1.3.1 Sampling lower bounds against codes

In our first main theorem, we show that it is hard to sample good codes using ROBPs. More generally, we obtain the following sampling lower bounds against any (n,k,d) code, which are nearly tight.

▶ **Theorem 1** (Sampling lower bounds against codes). *Let $\mathbf{Q} \sim \{0,1\}^n$ be uniform over an (n,k,d) code. Then for any ROBP $F : \{0,1\}^\ell \to \{0,1\}^n$ of width w,*

$$|F(\mathbf{U}_\ell) - \mathbf{Q}| \geq 1 - 12w \cdot 2^{-\frac{kd}{4n}}.$$

▶ Remark 2. We show that Theorem 1 is nearly tight, in the sense that for almost all "valid" parameters n,k,d, there exists an (n,k,d) code that can be sampled by an ROBP of width $2^{\widetilde{O}(\frac{kd}{n})}$. For a more formal statement of this result, we refer the reader to Section 5.

As a corollary, we immediately get that any distribution sampled by an ROBP of exponential width has statistical distance exponentially close to 1 from a good code, answering Question 3:

▶ **Corollary 3.** *For any good code* $\mathbf{Q} \sim \{0,1\}^n$*, there is a constant* $c > 0$ *such that for any ROBP* $F : \{0,1\}^\ell \to \{0,1\}^n$ *of width at most* 2^{cn}*,*

$$|F(\mathbf{U}_\ell) - \mathbf{Q}| \geq 1 - 12 \cdot 2^{-cn}.$$

Note that this is tight up to the constants c and 12, since statistical distance $\leq 1 - 2^{-n}$ is easily achieved by a width 1 ROBP that is constant over a single codeword. Finally, we note that we actually obtain a more general version of Theorem 1, which works for any list-decodable code: we refer the reader to Section 5 for more details.

We remark that these sampling lower bounds against codes for ROBPs are stronger than the best known sampling lower bounds against codes for AC^0. In particular, the best sampling lower bounds against good codes for AC^0 are of the form $1 - 2^{-n^{\Omega(1)}}$ [5], and the authors leave as an open problem whether similar lower bounds for AC^0 can be obtained against (n, k, d) codes with $kd \geq n^{1+\Omega(1)}$. On the other hand, our sampling lower bounds against good codes are of the form $1 - 2^{-\Omega(n)}$ for ROBPs of width $2^{\Omega(n)}$ (Corollary 3), and we obtain lower bounds of the form $1 - 2^{-n^{\Omega(1)}}$ against (n, k, d) codes with $kd \geq n^{1+\Omega(1)}$, for ROBPs of width $2^{n^{\Omega(1)}}$ (Theorem 1).

Applications to data structure lower bounds. By applying a known connection between sampling lower bounds and data structure lower bounds [22], we immediately get tight data structure lower bounds for storing codewords succinctly and retrieving them using ROBPs.

▶ **Corollary 4.** *For any good code* $Q \subseteq \{0,1\}^n$ *of dimension* k*, there is a constant* $c > 0$ *such that the following holds. If we can store codewords of* Q *using* $k + r$ *bits so that a codeword can be computed by an ROBP* $F : \{0,1\}^{k+r} \to \{0,1\}^n$ *of width at most* 2^{cn}*, then we must have redundancy* $r \geq \lfloor cn \rfloor$*.*

The above corollary shows that if one wishes to store codewords that are retrievable by a width $2^{\Omega(n)}$ ROBP, they must use $\Omega(n)$ extra bits of redundancy. This is tight up to constant factors: (1) It is easy to store codewords that are retrievable by a width 2^n ROBP using 0 extra bits of redundancy; and (2) It is easy to store codewords that are retrievable by a width 1 ROBP using $n - k$ extra bits of redundancy.

We remark that these data structure lower bounds for storing codes and retrieving them using ROBPs are stronger than the best known data structure lower bounds for storing codes and retrieving them using AC^0 circuits. In particular, for ROBPs of exponential width $2^{\Omega(n)}$, we show that $r \geq \Omega(n)$ bits of redundancy are necessary, whereas the best known result for AC^0 requires $r \geq n^{\Omega(1)}$ bits of redundancy.

1.3.2 A direct product theorem

Our second main theorem is a direct product theorem. This gives a generic way to construct distributions with strong sampling lower bounds against ROBPs. Informally, this theorem shows that if a distribution $\mathbf{Q}$ is even a little hard to sample for ROBPs, then the distribution $\mathbf{Q}^{\otimes t}$ (defined as a sequence of t independent copies of $\mathbf{Q}$) is extremely hard to sample for ROBPs. More formally, we prove the following.

▶ **Theorem 5** (Direct product theorem)**.** *Let* $\mathbf{Q} \sim \{0,1\}^n$ *be a distribution such that for any ROBP* $F : \{0,1\}^\ell \to \{0,1\}^n$ *of width* w*, it holds that* $|F(\mathbf{U}_\ell) - \mathbf{Q}| \geq \delta$*. Then for any* $t \in \mathbb{N}$ *and ROBP* $F^* : \{0,1\}^{\ell^*} \to \{0,1\}^{nt}$ *of width* w*, it holds that*

$$|F^*(\mathbf{U}_{\ell^*}) - \mathbf{Q}^{\otimes t}| \geq 1 - e^{-t\delta^2/8}.$$

In particular, our direct product theorem gives a way to boost statistical distance lower bounds of the form $\delta > 0$ (some tiny constant) to lower bounds of the form $1 - 2^{-\Omega(t)}$.

More generally, we actually obtain a direct product theorem that works for sampling with any computational model that has a certain "closure" property. We show that ROBPs exhibit such a property, and it is an interesting future direction to determine whether this is also true for other natural computational models. If such a closure property can be shown for AC^0, for example, then our general theorem would immediately imply a direct product theorem for sampling in AC^0. We refer the reader to Section 6 for more details.

A simple new lemma on amplifying statistical distance. A key ingredient in the proof of our direct product theorem is a simple new lemma on amplifying statistical distance between sequences of somewhat-dependent random variables. To the best of our knowledge, no such lemma was previously known, and we believe it may be of independent interest:

▶ **Lemma 6.** *Let $\mathbf{X} \sim V^n$ and $\mathbf{Y} \sim V^n$ each be a sequence of random variables over V, where elements in the sequence need not be independent. Suppose that for every $i \in [n]$ and $v \in V^{i-1}$,*

$$|(\mathbf{X}_i \mid \mathbf{X}_{<i} = v) - (\mathbf{Y}_i \mid \mathbf{Y}_{<i} = v)| \geq \delta.$$

Then

$$|\mathbf{X} - \mathbf{Y}| \geq 1 - e^{-n\delta^2/2}.$$

The proof of Lemma 6 is not difficult: we simply prove an analogous result over a more amenable notion of distance, known as the *Bhattacharyya coefficient*, and use estimates on statistical distance (in terms of the Bhattacharyya coefficient) to obtain the desired result. Despite its simple proof, we believe that it could be a useful tool for proving lower bounds. We discuss one such application, below.

Applications to sampling with two-party communication protocols. As an application of the above lemma, we obtain a simple new proof of a known result on sampling lower bounds for two-party communication protocols [14]. In particular, in Section 2.2, we provide a short, self-contained proof that for any distribution $\mathbf{X} \sim \{0,1\}^n \times \{0,1\}^n$ sampled by two-party communication protocols with $\Omega(n)$ bits of communication, it holds that $\mathbf{X}$ has statistical distance $1 - 2^{-\Omega(n)}$ from the distribution $\mathbf{Q} \sim \{0,1\}^n \times \{0,1\}^n$ that is uniform over pairs of disjoint strings.

Organization. The rest of this paper will be structured as follows. We start by giving an overview of our techniques in Section 2. In Section 3, we provide some basic preliminaries and facts that will be used throughout the paper. Next, in Section 4 we show how known (or easy-to-prove) communication lower bounds can be used to answer Question 1 and Question 2. Then, in Section 5, we obtain our sampling lower bounds against codes, proving our first main result (Theorem 1) and answering Question 3. In Section 6, we prove our direct product theorem for sampling with ROBPs (Theorem 5) and answer Question 4. We wrap up with some future directions in Section 7. We refer the reader to [11] for the full version of this paper, which contains all omitted proofs and several additional results.

2 Overview of our techniques

In this section, we give a detailed overview of the techniques that go into proving our two main theorems: namely, our sampling lower bounds against codes (Theorem 1), and our direct product theorem (Theorem 5). Before we dive into these proofs, we briefly discuss an important tool that we use throughout the paper, which helps streamline our arguments about sampling with ROBPs.

An equivalence between two small-space samplers. When all is said and done, our goal is to obtain theorems about sampling with ROBPs: that is, we wish to gain a deeper understanding of distributions of the form $F(\mathbf{U}_\ell)$, where $F : \{0,1\}^\ell \to \{0,1\}^n$ is a function computed by an ROBP. However, given the generality of ROBPs, this model of sampling can be a little cumbersome to work with formally.

In order to circumvent the need to work with this model directly, we will actually prove many of our results using a *different model* for sampling with limited memory, known as a *small-space source*. This model is much easier to work with formally, as its definition is simpler. Surprisingly, it also turns out that sampling with this model is roughly equivalent to sampling with ROBPs. This means that we can largely focus on proving results about the simpler model. We go into more detail below.

Small-space sources were introduced by Kamp, Rao, Vadhan and Zuckerman [18]. We henceforth refer to this model as the *KRVZ sampler*, and it is defined as a certain type of branching program that receives no input. More formally, a KRVZ sampler of width w and length n is a directed acyclic graph $G = (V, E)$ with layers $V = V_0 \cup V_1 \cup \cdots \cup V_n$, each holding w vertices. For every $i \in [n]$, each $v \in V_{i-1}$ can have an *arbitrary* number of edges into the next layer. The vertex v assigns a probability distribution p_v over its outgoing edges, and each of them also receive a label of 0 or 1. There is a distinguished start vertex $v_{\mathsf{start}} \in V_0$, and the KRVZ sampler generates a distribution $\mathbf{X} \sim \{0,1\}^n$ by taking a random walk from v_{start} according to the edge probabilities $\{p_v\}$, outputting all bits seen along the way.

Given this definition, we prove an equivalence theorem which says that a distribution $\mathbf{Q} \sim \{0,1\}^n$ is samplable by a KRVZ sampler of width w if and only if it is samplable by an ROBP of width w (ignoring a small loss in parameters: see Theorem 13). The more challenging direction of this proof is that KRVZ samplers $\implies$ ROBP samplers. Here, the difficulty is with simulating the multiple outgoing edges from each vertex, and the arbitrary probabilities it may assign over them. However, we show that such a simulation is possible by combining a lemma of [18] (to make the edge probabilities of the KRVZ sampler "granular") with a construction of a certain type of ROBP that sorts boolean strings into buckets of various sizes. In other words, the ROBP computes a type of "multi-thresholding" function.

Given this equivalence theorem, (1) Lower bounds against KRVZ samplers imply lower bounds against sampling with ROBPs; and (2) The existence of KRVZ samplers for a distribution implies that such a distribution can be sampled with ROBPs. As we will see now, both directions of this equivalence theorem will be useful in proving our main results.

2.1 Sampling lower bounds against codes

With our equivalence theorem in hand, we are ready to sketch the proof of our first main theorem (Theorem 1) and the proof of its tightness (Remark 2). Recall that Theorem 1 roughly says that for any distribution $\mathbf{Q} \sim \{0,1\}^n$ that is uniform over an (n, k, d) code

with good parameters (i.e., k, d, large), it holds that any distribution sampled by an ROBP (whose width is not too large) will have statistical distance close to 1 from $\mathbf{Q}$. Using our equivalence theorem, it suffices to prove this theorem for KRVZ samplers, instead.

In order to prove this theorem, we actually prove a more general version that works for list-decodable codes. Recall that a (ρ, L) list-decodable code of dimension k is a subset $Q \subseteq \{0,1\}^n$ of size 2^k such that any Hamming ball of radius $\leq \rho n$ contains at most L points from Q. Note that any (n, k, d) code Q is a $(\rho, 1)$ list-decodable code of dimension k, for any $\rho < \frac{d}{2n}$. Thus it suffices to show that for any distribution $\mathbf{Q}$ uniform over a list-decodable code with good parameters (i.e., k, ρ large, L small), it holds that any distribution sampled by a KRVZ sampler (whose width is not too large) will have statistical distance close to 1 from $\mathbf{Q}$.

The proof uses two main ingredients. First, it uses a known lemma [18] which says that any (distribution generated by a) KRVZ sampler $\mathbf{X} \sim \{0,1\}^n$ can be written as a convex combination of a few product distributions. More formally: if the sampler has width w, then for any r, ℓ with $r\ell = n$, it can be written as a convex combination of w^r distributions of the form $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_r) \sim (\{0,1\}^\ell)^r$, where each $\mathbf{Y}_i$ is independent.

The second ingredient, which we will prove, is that product distributions, i.e., those of the form $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_r) \sim (\{0,1\}^\ell)^r$ with each $\mathbf{Y}_i$ independent, are statistically far from (distributions that are uniform over) good list decodable codes.

At this point, it would be nice to conclude that the original KRVZ sampler $\mathbf{X}$ must also be far from a good list decodable code $\mathbf{Q}$. In particular, we would like to argue that $\mathbf{X}$ is a convex combination of product distributions $\{\mathbf{Y}^{(j)}\}$, and each of these product distributions is far from $\mathbf{Q}$, so $\mathbf{X}$ must be far from $\mathbf{Q}$. Unfortunately, the bounds we are trying to lift are in the wrong direction: it is true that if each $\mathbf{Y}^{(j)}$ is close to $\mathbf{Q}$, then $\mathbf{X}$ is close to $\mathbf{Q}$, but it is not necessarily true that if each $\mathbf{Y}^{(j)}$ is far from $\mathbf{Q}$, then $\mathbf{X}$ is far from $\mathbf{Q}$. Indeed, it could be the case that each $\mathbf{Y}^{(j)}$ is constant over a (different) codeword, which would make each $\mathbf{Y}^{(j)}$ extremely far from $\mathbf{Q}$, but still allow the overall convex combination over $\{\mathbf{Y}^{(j)}\}$ to exactly sample $\mathbf{Q}$.

Given the above counterexample, a new idea might be to try to argue that *as long as there aren't too many distributions* $\mathbf{Y}^{(j)}$ participating in the convex combination, then if each $\mathbf{Y}^{(j)}$ is far from $\mathbf{Q}$, then $\mathbf{X}$ is relatively far from $\mathbf{Q}$. It turns out this is true, but the corresponding lower bounds on $|\mathbf{X} - \mathbf{Q}|$ that it yields are still not as strong as we would like. To get the strongest possible bounds, we need a slightly more nuanced way to combine our two key ingredients.

Below, we sketch a proof for our second ingredient, and show to combine it with the first ingredient to yield our desired lower bound on $|\mathbf{X} - \mathbf{Q}|$.

Anti-concentration of product distributions in Hamming balls. We now argue that product distributions are far from sampling good list decodable codes. Let $\mathbf{Q} \sim \{0,1\}^n$ be a (ρ, L) list decodable code, and let $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_r) \sim (\{0,1\}^\ell)^r$ be such that each $\mathbf{Y}_i$ is independent and $r\ell = n$. Furthermore, we will need the product distribution to have a reasonable number of components r, or else it could clearly sample the code perfectly (if $r = 1$). Towards this end, we enforce the mild requirement $r \geq 1/\rho$, and thus $\ell \leq \rho n$. For a good list decodable code, we can think of $\rho = \Omega(1)$, and thus we just require $r \geq O(1)$.

Now, the key intuition about product distributions is that for any point x in the space $\{0,1\}^n$ to which $\mathbf{Y}$ does not assign too much probability, the following must hold: if we draw a Hamming ball $\mathcal{B}(x)$ around x whose radius is not too small, then the vast majority of probability weight assigned to $\mathcal{B}(x)$ by $\mathbf{Y}$ does *not* land on x. In symbols, $\Pr[\mathbf{Y} \in \mathcal{B}(x)] \gg \Pr[\mathbf{Y} = x]$.

Let us formalize this intuition a little more. Fix any $x \in \{0,1\}^n$, and let $p := \Pr[\mathbf{Y} = x]$. Consider now the ball $\mathcal{B}_\ell(x)$ around x of radius ℓ. Now, parse x as $x = (x_1, \ldots, x_r) \in (\{0,1\}^\ell)^r$. Since $\mathbf{Y}$ is a product distribution consisting of r components, there must be at least some $i \in [r]$ such that $\Pr[\mathbf{Y}_i = x_i] \leq p^{1/r}$. Consider now the set T of all strings of the form $(x_1, \ldots, x_{i-1}, z, x_{i+1}, \ldots, x_r) \in (\{0,1\}^\ell)^r$, where each x_j is fixed as before, but z can be taken as any element in $\{0,1\}^\ell$. Then $\mathbf{Y}$ assigns probability at least $\Pr[\mathbf{Y} = x]/p^{1/r}$ to the set T, and of course T is in the ball $\mathcal{B}_\ell(x)$. Thus, we get that $\Pr[\mathbf{Y} \in \mathcal{B}(x)] \geq \Pr[\mathbf{Y} = x]/p^{1/r}$, and therefore $\Pr[\mathbf{Y} \in \mathcal{B}(x)] \gg \Pr[\mathbf{Y} = x]$, as long as p is not too big.

Now, how can we use this to show $|\mathbf{Y} - \mathbf{Q}|$ is large? Well, by definition of statistical distance, it suffices to pick a set $S \subseteq \{0,1\}^n$ and show that $\Pr[\mathbf{Q} \in S] - \Pr[\mathbf{Y} \in S]$ is large. Our choice for S will be all codewords from $\mathbf{Q}$, with some "bad" codewords Bad removed. Then to lower bound $\Pr[\mathbf{Q} \in S]$, we just need to show that $\mathbf{Q}$ lands in Bad with not too high probability: in particular, we can just require that Bad is not too big. And to upper bound $\Pr[\mathbf{Y} \in S]$, we just need to show that the probability that $\mathbf{Y}$ lands on a "not-bad" codeword is small.

So what should we choose as the set Bad? You guessed it: a small set of codewords assigned the highest probability by $\mathbf{Y}$ (say, all codewords assigned probability $\geq p$ for some threshold probability p). As long as this set isn't too big (i.e., p isn't too small), we will have $\Pr[\mathbf{Q} \in S]$ be very close to 1. And as long as we removed the codewords assigned very high probability by $\mathbf{Y}$, we will have that $\Pr[\mathbf{Y} \in S]$ is very close to 0. We argue the latter, below.

To upper bound the probability that $\mathbf{Y}$ lands in S, we consider the sum $\sum_{q \in S} \Pr[\mathbf{Y} = q]$. By our anti-concentration observation above, this sum is at most $p^{1/r} \cdot \sum_{q \in S} \Pr[\mathbf{Y} \in \mathcal{B}_\ell(q)]$. Intuitively, we will now want to make sure that (i) r is not too big, because otherwise $p^{1/r}$ will be too big; and (ii) ℓ is not too big, because otherwise many of the balls $\{\mathcal{B}_\ell(q)\}$ in the sum will have big overlaps, causing probabilities to be multi-counted and the overall sum to be large.

It turns out that the best tradeoff occurs at setting $r = 1/\rho$ and $\ell = \rho n$. This is because a good list decodable code will have $\rho = \Omega(1)$, which yields $p^{1/r} = p^{\Omega(1)}$, which will be quite small as long as we originally set our threshold probability p to be low enough. Similarly, by definition of list decodability, we will have that any point x in the space $\{0,1\}^n$ will appear in at most L balls $\{\mathcal{B}_\ell(q)\}_{q \in S}$. This implies $\sum_{q \in S} \Pr[\mathbf{Y} \in \mathcal{B}_\ell(q)] \leq L$, since the probability $\mathbf{Y}$ assigns to any point $x \in \{0,1\}^n$ is counted at most L times. For a good list decodable code, L is quite small, and we finally have that $\Pr[\mathbf{Y} \in S] \leq p^{1/r} \cdot L$ will be very close to 0.

Thus, as long as our original product distribution $\mathbf{Y} \sim (\{0,1\}^\ell)^r$ had $r \approx 1/\rho$ and $\ell \approx \rho n$, we have a set $S \subseteq \{0,1\}^n$ that makes $\Pr[\mathbf{Q} \in S] - \Pr[\mathbf{Y} \in S]$ very close to 1, implying the statistical distance $|\mathbf{Y} - \mathbf{Q}|$ is very close to 1.

From KRVZ samplers to product distributions. Now, the question is: how do we use the fact that product distributions are far from good list decodable codes in order to argue that KRVZ samplers are far from list decodable codes? Well, let $\mathbf{X} \sim \{0,1\}^n$ be the KRVZ sampler, and $\mathbf{Q} \sim \{0,1\}^n$ be the list decodable code. We need to show that $|\mathbf{X} - \mathbf{Q}|$ is large. So we write $\mathbf{X}$ as a convex combination of at most w^r product distributions $\{\mathbf{Y}^{(j)}\}_j$, each of the form $\mathbf{Y}^{(j)} = (\mathbf{Y}_1^{(j)}, \ldots, \mathbf{Y}_r^{(j)}) \sim (\{0,1\}^\ell)^r$.

For any tester $S \subseteq \{0,1\}^n$, the statistical distance $|\mathbf{X} - \mathbf{Q}|$ is lower bounded by $\Pr[\mathbf{Q} \in S] - \Pr[\mathbf{X} \in S]$. Furthermore, it is easy to verify that this, in turn, is lower bounded by the worst $\Pr[\mathbf{Q} \in S] - \Pr[\mathbf{Y}^{(j)} \in S]$ (meaning the one that gives the smallest value). So what S should we pick?

From above, we know that as long as we set $r = 1/\rho$ and $\ell = \rho n$, then each $\mathbf{Y}^{(j)}$ has a test $S^{(j)}$ which makes $\Pr[\mathbf{Y}^{(j)} \in S^{(j)}]$ very close to 0. Furthermore $S^{(j)}$ is of the form $Q - \mathsf{Bad}^{(j)}$, where Q is the support of the code and $\mathsf{Bad}^{(j)}$ is some small bad set. Thus, since we want a *single* test $S \subseteq \{0,1\}^n$ guaranteed to make $\Pr[\mathbf{Q} \in S] - \Pr[\mathbf{Y}^{(j)} \in S]$ small *for every* j, we can simply take the test to be $S = Q - \cup_j \mathsf{Bad}^{(j)}$. Indeed, this guarantees that each $\Pr[\mathbf{Y}^{(j)} \in S]$ will be very close to 0, and as long as there are not too many elements in the convex combination, our total collection of bad elements won't be too big, and $\Pr[\mathbf{Q} \in S]$ will stay very close to 1. Thus we get that $|\mathbf{X} - \mathbf{Q}|$ is close to 1, as desired.

On the tightness of our result. Above, we sketched the proof that any KRVZ sampler of not-too-large width generates a distribution that is very far from a good (n, k, d) code. More precisely, our result (Theorem 1) says that any KRVZ sampler of width $2^{\Omega(\frac{kd}{n})}$ has statistical distance $1 - 2^{-\Omega(\frac{kd}{n})}$ from any (n, k, d) code. In a complementary result, we show that this is nearly tight.

In more detail, we show that for almost all "valid" n, k, d, there is an (n, k, d) code that can be sampled by a KRVZ sampler (and thus an ROBP sampler) of width $2^{O(\frac{kd}{n} \cdot \log n)}$. More formally, we show this tightness result for all n, k, d for which there exists a linear (n, k, d) code.

Our proof of tightness is split into two cases. In the first case, we consider $k \leq 0.9n$. Here, the general idea is to define some n', k', d' such that there exists an (n', k', d') code Q', and then simply consider the repetition code $Q := Q' \times Q' \times \cdots \times Q'$, where n/n' copies of Q' participate in the Cartesian product. It is straightforward to verify that Q will be an $(n, k'n/n', d')$ code. Furthermore, it is not hard to see that a KRVZ sampler of width w can sample any distribution with support size w, and that the product distribution of two distributions, each samplable by a KRVZ sampler of width w, is also samplable by a KRVZ sampler of width w. Thus, Q will be samplable by a KRVZ sampler of width $2^{k'}$. Thus, if we can find a constant C and an (n', k', d') code with $n' = Cd, k' = Ckd/n, d' = d$, we are done with this case. Since $k \leq 0.9n$, the Gilbert-Varshamov bound guarantees this is always possible.

In the second case, we consider $k > 0.9n$. Here, the general idea is that $n - k$ will now be small. We consider two subcases: $k \geq n - 4d \log n$ and $k < n - 4d \log n$. We focus on the first subcase in this overview, as it is not too hard to extend the argument to work for the second subcase. In the first subcase, note that $n - k \leq 4d \log n$. Now, the main idea is that ROBPs can check membership of a k dimensional subspace $Q \subseteq \mathbb{F}_2^n$ using width 2^{n-k}, simply by keeping track of the $n - k$ parity checks that define Q. Furthermore, it is not too hard to show that for any ROBP of width w, the uniform distribution over its accepting strings can be generated by a KRVZ sampler of width w. Thus in this case, we can simply take any linear (n, k, d) code and uniformly sample from it using a KRVZ sampler of width $2^{n-k} \leq 2^{4d \log n} \leq 2^{\frac{5kd}{n} \cdot \log n}$, where the last inequality follows from the current case $k > 0.9n$.

2.2 A direct product theorem

We now begin our sketch of our second main theorem, Theorem 5. Recall that it roughly says that if a distribution $\mathbf{Q} \sim \{0,1\}^n$ has statistical distance $\geq \delta$ from distributions sampled by ROBPs of width w, then $\mathbf{Q}^{\otimes t} \sim \{0,1\}^{nt}$ has statistical distance $\geq 1 - 2^{-\Omega(t\delta^2)}$ from distributions sampled by ROBPs of width w. Here, recall that $\mathbf{Q}^{\otimes t}$ refers to a sequence of t independent copies of $\mathbf{Q}$.

In order to prove the above direct product theorem, we start by proving an analogous result for KRVZ samplers. Then, we use our equivalence theorem to obtain a direct product theorem for sampling with ROBPs. However, since our equivalence theorem (Theorem 13)

has some loss in parameters (width), this will only yield a *weak* direct product theorem: in such a result, the statistical distance still blows up from δ to $1 - 2^{-\Omega(t\delta^2)}$, *but only if* we also require the width to *decrease* from w to $w/14$.

We would really like a *strong* direct product theorem, in the sense that the statistical distance blows up even if the ROBP is allowed to keep all of its width w. At the end of this subsection, we show how to build some extra machinery to make this happen.

A direct product theorem for KRVZ samplers. We now proceed to sketch the proof of our direct product theorem for KRVZ samplers.

Let $\mathbf{X} \sim \{0,1\}^{nt}$ be a distribution generated by a KRVZ sampler of width w, and parse it as $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_t)$, where each $\mathbf{X}_i \sim \{0,1\}^n$ need not be independent. Recall that $\mathbf{Q}^{\otimes t} \sim \{0,1\}^{nt}$ is of the form $\mathbf{Q}^{\otimes t} = (\mathbf{Q}_1, \ldots, \mathbf{Q}_t)$, where each $\mathbf{Q}_i \sim \{0,1\}^n$ is an independent copy of $\mathbf{Q}$. We would like to argue

$$|(\mathbf{X}_1, \ldots, \mathbf{X}_t) - (\mathbf{Q}_1, \ldots, \mathbf{Q}_t)| \geq 1 - 2^{-\Omega(t\delta^2)}, \tag{1}$$

given that for any KRVZ sampler $\mathbf{X}' \sim \{0,1\}^n$ of width w it holds that $|\mathbf{X}' - \mathbf{Q}| \geq \delta$.

The first observation is that any of the $\mathbf{X}_i \sim \{0,1\}^n$ can be generated by a KRVZ sampler of width w. Indeed, even though it represents a sequence of bits generated in the middle of the KRVZ sampler $\mathbf{X}$, it is easy to create a new KRVZ sampler $\mathcal{B}$ of the same width that only generates $\mathbf{X}_i$, simply by: (1) copying the KRVZ sampler that creates $\mathbf{X}$; (2) throwing out all layers that do not produce bits corresponding to $\mathbf{X}_i$; (3) adding a new start vertex v_{start}; (4) connecting that start vertex the first layer remaining in $\mathcal{B}$, using the appropriate probabilities; and (5) merging the first two layers of $\mathcal{B}$, to deal with the fact that the edges leaving v_{start} currently have no output labels.

Thus, we are guaranteed that for each $\mathbf{X}_i \sim \{0,1\}^n$ and $\mathbf{Q}_i \sim \{0,1\}^n$, it holds that $|\mathbf{X}_i - \mathbf{Q}_i| \geq \delta$ by the theorem hypothesis (that every KRVZ sampler of width w is far from $\mathbf{Q}$). The question now is: is this enough to guarantee that the statistical distance blows up in Equation (1)?

Well, if each $\mathbf{X}_i$ were independent, it is not too hard to show that the answer is yes. However, this is of course not guaranteed to be the case, since the $\mathbf{X}_i$'s are consecutive slices of the same KRVZ sampler $\mathbf{X}$. Indeed, without further examination, it could potentially be the case that for any $x \in \{0,1\}^n$ and $i \in [n]$, the distributions $(\mathbf{X}_{-i} \mid \mathbf{X}_i = x)$ and $(\mathbf{Q}_{-i}^{\otimes t})$ are identical.[1] In this case, we cannot hope to lower bound Equation (1) by anything more than δ. In some sense, the above adversarial example represents a situation where each $\mathbf{X}_i$ is *not* contributing its "fair share" to the statistical distance in Equation (1). In order to force each $\mathbf{X}_i$ to be a contributing member, we would like a different guarantee than just $|\mathbf{X}_i - \mathbf{Q}_i| \geq \delta$. One natural way to encode the idea that each $\mathbf{X}_i$ is contributing its fair share is to require that for every $i \in [t]$ and $x \in (\{0,1\}^n)^{i-1}$,

$$|(\mathbf{X}_i \mid \mathbf{X}_1, \ldots, \mathbf{X}_{i-1} = x) - (\mathbf{Q}_i \mid \mathbf{Q}_1, \ldots, \mathbf{Q}_{i-1} = x)| \geq \delta. \tag{2}$$

This leaves us with two questions: (i) Given a guarantee like Equation (2), can we actually prove Equation (1)? (ii) Is the guarantee given in Equation (2) even true? If we can answer both questions in the affirmative, then we are done with our direct product theorem for KRVZ samplers.

[1] For a random variable $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_t)$, the notation $\mathbf{X}_{-i}$ denotes $\mathbf{X}$ with $\mathbf{X}_i$ removed.

It turns out that (i) is true, but it is a little cumbersome to do so using statistical distance. To avoid this, we use simple and well known facts to convert the statement into one about squared Hellinger distance, which can further be phrased in terms of the *Bhattacharyya coefficient*. Phrasing (i) in this way allows for a simple inductive proof, which we can then convert back to a result about statistical distance.

It also turns out that (ii) is true. To see why, note that $(\mathbf{Q}_i \mid \mathbf{Q}_1, \dots, \mathbf{Q}_{i-1} = x)$ is just the same distribution as $\mathbf{Q} \sim \{0,1\}^n$, since each $\mathbf{Q}_i$ is an independent copy of $\mathbf{Q}$. Furthermore, it is straightforward to show that the distribution $(\mathbf{X}_i \mid \mathbf{X}_1, \dots, \mathbf{X}_{i-1} = x)$ can be generated by a width w KRVZ sampler, using a similar idea to the one we presented for why $\mathbf{X}_i$ has this property. Thus the hypothesis of the direct product theorem implies Equation (2), and our direct product theorem for KRVZ samplers is complete.

A direct product theorem for sampling with ROBPs. It is now easy to obtain a direct product theorem for ROBP samplers, in a black-box manner, by combining the above direct product theorem with our equivalence theorem between KRVZ samplers and ROBP samplers (Theorem 13). However, as discussed at the beginning of this section, this will only yield a weak direct product theorem, since our equivalence theorem suffers a slight loss in parameters (i.e., width). If we want to obtain a *strong* direct product theorem for ROBP samplers, we must dig into the black box.

Looking back at the previous discussion, it is not too difficult to see that if one wants a strong direct product theorem for ROBP samplers (that suffers no loss in width), then it suffices to show the following: for any distribution $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_t) \sim (\{0,1\}^n)^t$ sampled by an ROBP of width w, and any $i \in [t], x \in (\{0,1\}^n)^{i-1}$, the distribution $(\mathbf{X}_i \mid \mathbf{X}_1, \dots, \mathbf{X}_{i-1} = x)$ can be sampled by an ROBP of width w.

In order to show the above, our key ingredient is the following: for any w and probability distribution $p : [w] \to \mathbb{R}_{\geq 0}$, we construct an ROBP of width w such that a random walk over it hits the i^{th} vertex in the last layer with probability $p(i) + \gamma$, where $\gamma > 0$ can be arbitrarily small. A first attempt at constructing such an ROBP might use the "multi-thresholding" discussed at the beginning of Section 2, which was used in our equivalence theorem. However, our construction of such a function required width $2w$ (instead of the desired w), and we show that this is tight up to additive constants.

To get the width down to w, we start by showing that for any biased coin $\mathbf{A} \sim \{0,1\}$, there is an ROBP of width 2 that samples a distribution arbitrarily close to it. At a high level, this argument works as follows: for any binary string $b \in \{0,1\}^\ell$, we show how to use its bits as "instructions" to construct a certain ROBP of length ℓ and width 2 in a layer-by-layer fashion. The constructed ROBP then guarantees that it accepts a random string with probability b, where b is interpreted as the binary representation of a number in $[0,1]$. Finally, it is not too hard to bootstrap such an object to create our desired ROBP of width w that hits the vertices in its last layer with probabilities close to $\{p(i)\}_{i \in [w]}$. As a result, we get our strong direct product theorem for sampling with ROBPs.

A key ingredient of our direct product theorem, and a simple new proof of [14]. A key ingredient of the above proofs is a simple new lemma on amplifying statistical distance between sequences of somewhat-dependent random variables. In particular, we show that for any random variables $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_n)$ and $\mathbf{Q} = (\mathbf{Q}_1, \dots, \mathbf{Q}_n)$ over the same domain, if $|(\mathbf{X}_i \mid \mathbf{X}_{<i} = x) - (\mathbf{Q}_i \mid \mathbf{Q}_{<i} = x)| \geq \delta$ for each i, x, then $|\mathbf{X} - \mathbf{Q}| \geq 1 - 2^{-\Omega(n\delta^2)}$. The proof is not difficult, and we believe it could be a useful new tool for proving lower bounds.

As an application, we give a simple new proof of a result by Göös and Watson [14] that any distribution $(\mathbf{X}, \mathbf{Y}) \sim \{0,1\}^n \times \{0,1\}^n$ sampled by two-party communication protocols (with communication $b = \Omega(n)$) is far from the distribution $(\mathbf{A}, \mathbf{B}) \sim \{0,1\}^n \times \{0,1\}^n$ that is uniform over pairs of disjoint strings. We present it below:

1. First, use the standard observation (made in, e.g., [2]) that $(\mathbf{X}, \mathbf{Y})$ is a convex combination of 2^b product distributions $(\mathbf{X}', \mathbf{Y}')$.
2. Use a standard data processing inequality to observe that

$$|(\mathbf{X}', \mathbf{Y}') - (\mathbf{A}, \mathbf{B})| \geq |(\mathbf{X}'_1, \mathbf{Y}'_1, \ldots, \mathbf{X}'_n, \mathbf{Y}'_n) - (\mathbf{A}_1, \mathbf{B}_1, \ldots, \mathbf{A}_n, \mathbf{B}_n)|.$$

3. Using a straightforward calculation, observe that $|(\mathbf{X}'_i, \mathbf{Y}'_i) - (\mathbf{A}_i, \mathbf{B}_i)| \geq \Omega(1)$ for each i, since $\mathbf{X}'_i, \mathbf{Y}'_i$ are independent and $(\mathbf{A}_i, \mathbf{B}_i)$ is uniform over $\{(0,0), (0,1), (1,0)\}$. Observe that this still holds even if you condition on any fixing of the random variables earlier on in the sequence, since this doesn't break the independence of $\mathbf{X}'_i, \mathbf{Y}'_i$, nor does it change the distribution of $(\mathbf{A}_i, \mathbf{B}_i)$.
4. Use our new lemma on amplifying statistical distance to conclude $|(\mathbf{X}', \mathbf{Y}') - (\mathbf{A}, \mathbf{B})| \geq 1 - 2^{-\Omega(n)}$.
5. Using the fact that the convex combination of a few far distributions is still far (Lemma 8), conclude that $|(\mathbf{X}, \mathbf{Y}) - (\mathbf{A}, \mathbf{B})| \geq 1 - 2^{b-\Omega(n)}$, which is $1 - 2^{-\Omega(n)}$ for some $b = \Omega(n)$, as desired.

3 Preliminaries

Before we start our formal proofs, we introduce some basic notation, definitions, and facts.

General notation. For any natural number $n \in \mathbb{N}$, we let $[n]$ denote the set $\{1, 2, \ldots, n\}$. Given a string $x \in \{0,1\}^n$ and index $i \in [n]$, we let x_i denote the i^{th} coordinate of x. Furthermore, for any $1 \leq i \leq j \leq n$, we let $x_{i \to j} := (x_i, \ldots, x_j) \in \{0,1\}^{j-i+1}$, we let $x_{\leq i} := x_{1 \to i}$, and we let $x_{<i} := x_{1 \to i-1}$.

Basic probability definitions and notation. We let $\mathbf{U}_n$ denote the uniform random variable over $\{0,1\}^n$. When $\mathbf{U}_n$ appears in the same expression twice, it denotes the same random variable - that is, they are *not* independent. Throughout, we slightly abuse notation and let $\mathbf{X} \sim \{0,1\}^n$ denote both a random variable and its underlying distribution. As is standard, we measure the distance between two distributions using statistical distance:

▶ **Definition 7.** *The* statistical distance *between two discrete random variables* $\mathbf{X}, \mathbf{Y} \sim V$ *is*

$$|\mathbf{X} - \mathbf{Y}| := \max_{S \subseteq V} |\Pr[\mathbf{X} \in S] - \Pr[\mathbf{Y} \in S]| = \frac{1}{2} \sum_{x \in V} |\Pr[\mathbf{X} = x] - \Pr[\mathbf{Y} = x]|.$$

We say that $\mathbf{X}, \mathbf{Y}$ are ε-close if the statistical distance between them is at most ε, and we say they are ε-far otherwise. A useful tool for bounding statistical distance is the *data processing inequality*, which says that for any discrete $\mathbf{X}, \mathbf{Y} \sim V$ and any function $f : V \to W$, it holds that $|f(\mathbf{X}) - f(\mathbf{Y})| \leq |\mathbf{X} - \mathbf{Y}|$.

Next, we say that $\mathbf{X}$ is a convex combination of distributions $\mathbf{Y}_1, \ldots, \mathbf{Y}_k$ if $\mathbf{X} = \sum_i p_i \mathbf{Y}_i$, for some probabilities $\{p_i\}$ that sum to 1. That is, $\mathbf{X}$ samples from $\mathbf{Y}_i$ with probability p_i. Finally, in the full version of the paper we prove the following lemma, which slightly generalizes a result of Viola [26].

▶ **Lemma 8.** *Let $\mathbf{X}, \mathbf{Q}$ be any two random variables over the same discrete space V. Suppose that $\mathbf{X}$ is a convex combination of t distributions $\mathbf{X} = \sum_{i\in[t]} p_i \mathbf{Y}_i$, where for each $\mathbf{Y}_i$ we have $|\mathbf{Y}_i - \mathbf{Q}| \geq 1 - \delta$. Then*

$$|\mathbf{X} - \mathbf{Q}| \geq 1 - t\delta.$$

Basic coding theory definitions and facts. Given two points $x, y \in \{0,1\}^n$, the Hamming distance $\Delta(x, y)$ between x, y is the number of coordinates where they differ. Next, given a point $x \in \{0,1\}^n$, the Hamming ball $\mathcal{B}_r(x)$ centered at x with radius r is the collection of points in $\{0,1\}^n$ that are Hamming distance at most r from x. An (n, k, d) code $Q \subseteq \{0,1\}^n$ is a collection of 2^k points such that the minimum Hamming distance between any two points is d. We call k its dimension and d its distance, and we say that Q is a linear $[n, k, d]$ code if it is also a subspace of $\mathbb{F}_2^n$.

A subset $Q \subseteq \{0,1\}^n$ is a (ρ, L) list decodable code if every Hamming ball in $\{0,1\}^n$ of radius at most ρn contains at most L points from Q. Notice that list decodable codes relax the distance requirement of (n, k, d) codes. Furthermore, it is straightforward to use the triangle inequality to show the following.

▶ **Fact 1.** *If $Q \subseteq \{0,1\}^n$ is an (n, k, d) code, then Q is $(\rho, 1)$ list decodable for any $\rho < \frac{d}{2n}$.*

3.1 Models for computing in small space

Read-once branching programs (ROBPs) are a popular model for computation in small space. We provide their standard definition, below.

▶ **Definition 9** (ROBP). *An* ROBP *$\mathcal{B}$ of width w and length n is a directed acyclic graph $G = (V, E)$ consisting of $n+1$ disjoint layers $V = V_0 \cup V_1 \cup \cdots \cup V_n$, each holding w vertices. For every $i \in [n]$, each vertex $v \in V_{i-1}$ has exactly two outgoing edges into V_i, one of which is labeled* 0*, and the other labeled* 1*. There is a designated* start vertex $v_{\mathsf{start}} \in V_0$*, and a designated* accept vertex $v_{\mathsf{accept}} \in V_n$*.*

The branching program $\mathcal{B}$ computes a function $f_{\mathcal{B}} : \{0,1\}^n \to \{0,1\}$ as follows: on input $x \in \{0,1\}^n$, the program starts at v_{start} and traverses the unique path $P(x)$ whose edges are labeled with input bits $x_1, x_2, \ldots, x_n$. The program outputs 1 *if $P(x)$ terminates on v_{accept}, and* 0 *otherwise.*

As motivated in Section 1.2, we use the following definition for multi-output ROBPs.

▶ **Definition 10** (Multi-output ROBP). *A* multi-output ROBP *$\mathcal{B}$ of width w and (input) length n is a directed acyclic graph $G = (V, E)$ consisting of $n+1$ disjoint layers $V = V_0 \cup V_1 \cup \cdots \cup V_n$, each holding w vertices. For every $i \in [n]$, each vertex $v \in V_{i-1}$ has exactly two outgoing edges into V_i, one of which is labeled with the input bit* 0*, and the other labeled with the input bit* 1*. Each edge e is also labeled with* output bits *$\Gamma(e) \in \{0,1\}^*$, and we assume that all edges e between the same two layers V_{i-1}, V_i have the same output length $|\Gamma(e)| = \gamma_i \geq 0$. The* output length *of $\mathcal{B}$ is $m = \sum_i \gamma_i$. Finally, there is a designated* start vertex $v_{\mathsf{start}} \in V_0$*.*

The branching program $\mathcal{B}$ computes a function $f_{\mathcal{B}} : \{0,1\}^n \to \{0,1\}^m$ as follows: on input $x \in \{0,1\}^n$, the program starts at v_{start} and traverses the unique path $P(x)$ whose edges are labeled with input bits $x_1, x_2, \ldots, x_n$. The program outputs the concatenation of all output bits seen along this path, so that $f_{\mathcal{B}}(x) = (\Gamma(e))_{e\in P(x)}$.

It is straightforward to verify that Definition 10 is a strict generalization of Definition 9. Because of this, we will omit the qualifier "multi-output" when referring to ROBPs.

3.2 Models for sampling in small space

We now introduce our models for sampling in small space. We start with main motivating model of simply feeding uniform bits into an ROBP:

▶ **Definition 11** (ROBP sampler). *An* ROBP sampler $\mathcal{B}$ *of width* w *and input length* ℓ *and output length* n *is just an ROBP with the same parameters. The distribution* $\mathbf{X} \sim \{0,1\}^n$ *sampled by the ROBP* $\mathcal{B}$ *is* $\mathbf{X} = f_{\mathcal{B}}(\mathbf{U}_\ell)$*, where* $f_{\mathcal{B}}$ *is the function computed by* $\mathcal{B}$*.*

The next type of sampler was defined by by Kamp, Rao, Vadhan, and Zuckerman under the name of *small space sources* [18]; we will henceforth call it a KRVZ sampler.

▶ **Definition 12** (KRVZ sampler). *A* KRVZ sampler $\mathcal{B}$ *of width* w *and output length* n *is a directed acyclic graph* $G = (V, E)$ *consisting of* $n+1$ *disjoint layers* $V = V_0 \cup V_1 \cup \cdots \cup V_n$*, each holding* w *vertices. For every* $i \in [n]$*, each vertex* $v \in V_{i-1}$ *has an* arbitrary number of outgoing edges *into* V_i*, some of which are labeled* 0*, and the rest labeled* 1*. There is a designated start vertex* $v_{\mathsf{start}} \in V_0$*, and each vertex* $v \in V$ *has a probability distribution* p_v *over its outgoing edges. The distribution* $\mathbf{X} \sim \{0,1\}^n$ *sampled by* $\mathcal{B}$ *is the one generated by taking a random walk over* $\mathcal{B}$*, which starts at* v_{start}*, transitions according to* $\{p_v\}$*, and outputs the edge labels seen along the way.*

It turns out that ROBP samplers and KRVZ samplers are roughly equivalent. We prove the following equivalence theorem in the full version of the paper.

▶ **Theorem 13** (Equivalence theorem). *For any distribution* $\mathbf{X} \sim \{0,1\}^n$*:*

- *If there is an ROBP of width* w *and input length* ℓ *that samples* $\mathbf{X}$*, then there exists a KRVZ sampler of width* $2w$ *that samples* $\mathbf{X}$*.*
- *If there exists a KRVZ sampler of width* w *that samples* $\mathbf{X}$*, then for any* $\varepsilon > 0$*, there exists an ROBP of width* $7w$ *and input length* $\ell = 8nw \log(nw/\varepsilon)$ *that samples a distribution that is* ε*-close to* $\mathbf{X}$*.*

We say that a KRVZ sampler is α*-granular* if each edge probability is an integer multiple of α, and for such samplers we strengthen the second bullet of Theorem 13 as follows.

▶ **Lemma 14.** *For any distribution* $\mathbf{X} \sim \{0,1\}^n$*, if there exists a* (2^{-t})*-granular KRVZ sampler of width* w *that samples* $\mathbf{X}$*, then there exists an ROBP of width* $7w$ *and input length* $\ell = 4nwt$ *that samples* $\mathbf{X}$*.*

We also prove a different version of this lemma, which focuses on minimizing the randomness used by the ROBP sampler, at the expense of introducing slightly more width.

▶ **Lemma 15.** *For any distribution* $\mathbf{X} \sim \{0,1\}^n$*, if there exists a* (2^{-t})*-granular KRVZ sampler of width* w *that samples* $\mathbf{X}$*, then there exists an ROBP of width* $4w^2$ *and input length* $\ell = nt$ *that samples* $\mathbf{X}$*.*

In the full version of the paper, we give two more equivalence theorems that relate *sampling* in small space to *computing* in small space. These theorems can be used to obtain correlation bounds from sampling lower bounds, in addition to other applications.

4 Results that are easy to obtain using communication complexity

As it turns out, many natural questions about sampling with ROBPs can be answered using known (or easy-to-prove) results from communication complexity. We briefly overview these results here, and refer the reader to the full version for more details and proofs.

The first result is that *sampling is easier than computing* for ROBPs, answering Question 1.

▶ **Theorem 16** (Sampling is easier than computing). *There exists an explicit function* $b : \{0,1\}^n \to \{0,1\}$ *such that for any* $\varepsilon > 0$*, the following holds. For every ROBP* $F : \{0,1\}^n \to \{0,1\}$ *of width at most* $2^{\frac{n\varepsilon^2}{9} - \log(1/\varepsilon)}$ *it holds that*

$$\Pr_x[F(x) = b(x)] < \frac{1}{2} + \varepsilon,$$

but there exists an ROBP $G : \{0,1\}^\ell \to \{0,1\}^{n+1}$ *of width* $2n$ *(and length* $\ell \leq n + \log n + 2$*) such that*

$$G(\mathbf{U}_\ell) = (\mathbf{U}_n, b(\mathbf{U}_n)).$$

To prove this result, we take $b : \{0,1\}^n \to \{0,1\}$ to be the function $\mathsf{address} : \{0,1\}^k \times [k] \to \{0,1\}$, defined as $\mathsf{address}(x,i) := x_i$, where $n = k + \log k$. The theorem then follows almost immediately from known communication lower bounds against the address function. We thank an anonymous reviewer for a concise proof of these communication lower bounds, which are included in the full version.

The second result is *sampling lower bounds against input-output pairs*, thereby answering Question 2.

▶ **Theorem 17** (Sampling lower bounds against input-output pairs). *There exists a universal constant* $c > 0$ *and an explicit function* $b : \{0,1\}^n \to \{0,1\}$ *such that for any* $\ell = \ell(n)$ *and ROBP* $F : \{0,1\}^\ell \to \{0,1\}^{n+1}$ *of width at most* 2^{cn}*,*

$$|F(\mathbf{U}_\ell) - (\mathbf{U}_n, b(\mathbf{U}_n))| \geq \frac{1}{2} - 2^{-cn}.$$

To prove this result, we take $b : \{0,1\}^n \to \{0,1\}$ to be a sufficiently good *two-source extractor*. It is then not too hard to show that communication protocols have a very hard time sampling $(\mathbf{U}_n, b(\mathbf{U}_n))$, from which Theorem 17 follows immediately (since communication protocols can simulate ROBPs). As a bonus, we get a separation between sampling with ROBPs and AC^0 circuits for *input-output pairs*. In particular, we see that for the inner product function $\mathsf{IP} : \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$, the input-output distribution $(\mathbf{U}_n, \mathsf{IP}(\mathbf{U}_n))$ cannot be sampled by exponential width ROBPs even on average, since IP is a good two-source extractor, but $(\mathbf{U}_n, \mathsf{IP}(\mathbf{U}_n))$ can be sampled in AC^0, by a result of Impagliazzo and Naor [16].

The third result is *a very simple distribution that is very hard to sample for ROBPs*.

▶ **Theorem 18** (Simple distributions that are hard to sample). *For any* $\ell \in \mathbb{N}$ *and ROBP* $F : \{0,1\}^\ell \to \{0,1\}^{2n}$ *of width at most* $2^{n/6}$*,*

$$|F(\mathbf{U}_\ell) - (\mathbf{U}_n, \mathbf{U}_n)| \geq 1 - 16 \cdot 2^{-n/6}.$$

To prove this result, we simply recall that communication protocols can only sample convex combinations of product distributions, each of which is far from $(\mathbf{U}_n, \mathbf{U}_n)$. We then use Lemma 8 to show that communication protocols have a hard time sampling $(\mathbf{U}_n, \mathbf{U}_n)$, from which Theorem 18 follows immediately (since communication protocols can simulate ROBPs). As a bonus, we get an even stronger separation between sampling with ROBPs and AC^0 circuits, since AC^0 can clearly sample the distribution $(\mathbf{U}_n, \mathbf{U}_n)$.

5 Sampling lower bounds against codes

In this section, we prove our first main theorem (Theorem 1) on sampling lower bounds against codes. We start by proving a more general result (for KRVZ samplers).

▶ **Theorem 19.** *Let $\mathbf{Q} \sim \{0,1\}^n$ be uniform over a (ρ, L) list decodable code of dimension k. Then for any KRVZ sampler $\mathbf{X} \sim \{0,1\}^n$ of width w,*

$$|\mathbf{X} - \mathbf{Q}| \geq 1 - 4wL \cdot 2^{-\frac{\rho}{1+2\rho}k}.$$

Proof. We start by writing our KRVZ sampler as a convex combination of distributions with nice structure. Let $\mathbf{W} = (\mathbf{W}_1, \ldots, \mathbf{W}_n)$ be the vertices hit on the random walk that generates $\mathbf{X}$. Let r, ℓ be positive integers that will be set later to ensure $r\ell = n$. Define $\mathbf{W}^* = (\mathbf{W}_\ell, \mathbf{W}_{2\ell}, \ldots, \mathbf{W}_{r\ell})$, and recall the following observation [19, 20, 18]: for any $W \in \text{support}(\mathbf{W}^*)$, the random variable $(\mathbf{X} \mid \mathbf{W}^* = W)$ is of the form $\mathbf{X}^{(W)} := (\mathbf{X}_1^{(W)}, \mathbf{X}_2^{(W)}, \ldots, \mathbf{X}_r^{(W)})$, where each $\mathbf{X}_i^{(W)} \sim \{0,1\}^\ell$ is independent. Thus the KRVZ sampler $\mathbf{X}$ is a convex combination of the form

$$\mathbf{X} = \sum_{W \in \text{support}(\mathbf{W}^*)} p_W \cdot \mathbf{X}^{(W)},$$

where each $p_W := \Pr[\mathbf{W}^* = W]$.

The goal now is to use the above decomposition to help us get a good lower bound on $|\mathbf{X} - \mathbf{Q}| = \max_S |\Pr[\mathbf{X} \in S] - \Pr[\mathbf{Q} \in S]|$. Towards this end, we note that for any S,

$$|\mathbf{X} - \mathbf{Q}| \geq \Pr[\mathbf{Q} \in S] - \Pr[\mathbf{X} \in S] \geq \Pr[\mathbf{Q} \in S] - \max_W \Pr[\mathbf{X}^{(W)} \in S].$$

Next, let $t > 0$ be a parameter to be set later, and we define $S = Q - \mathsf{Bad}$ where $\mathsf{Bad} := \bigcup_W \mathsf{Bad}^{(W)}$ and $\mathsf{Bad}^{(W)} := \{q \in Q : \Pr[\mathbf{X}^{(W)} = q] > 2^{-t}\}$. Plugging in this definition of S,

$$|\mathbf{X} - \mathbf{Q}| \geq 1 - \Pr[\mathbf{Q} \in \mathsf{Bad}] - \max_W \Pr[\mathbf{X}^{(W)} \in Q - \mathsf{Bad}^{(W)}].$$

We would like to upper bound both quantities that are subtracted. To upper bound the first quantity, simply note that

$$\Pr[\mathbf{Q} \in \mathsf{Bad}] = 2^{-k} \cdot |\mathsf{Bad}| \leq 2^{-k} \sum_{W \in \text{support}(\mathbf{W}^*)} |\mathsf{Bad}^{(W)}| < 2^{-k+t+r\log(w)}$$

via the trivial upper bounds $|\text{support}(\mathbf{W}^*)| \leq w^r$ and $|\mathsf{Bad}^{(W)}| < 2^t$ for each W.

To upper bound the second quantity $\max_W \Pr[\mathbf{X}^{(W)} \in Q - \mathsf{Bad}^{(W)}]$, we start by making notation more convenient: let W^* be the maximizer of the above quantity, and define $\mathbf{Y} := \mathbf{X}^{(W^*)}$ and $\mathsf{Bad}^* := \mathsf{Bad}^{(W^*)}$. Of course we have $\max_W \Pr[\mathbf{X}^{(W)} \in Q - \mathsf{Bad}^{(W)}] = \Pr[\mathbf{Y} \in Q - \mathsf{Bad}^*]$, and we focus on upper bounding the latter.

Recall that $\mathbf{Y}$ is of the form $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_r)$ where each $\mathbf{Y}_i \sim \{0,1\}^\ell$ is independent, and $\mathsf{Bad}^* := \{q \in Q : \Pr[\mathbf{Y} = q] > 2^{-t}\}$ contains all codewords hit by $\mathbf{Y}$ with large probability. Thus each $q \in Q - \mathsf{Bad}^*$ must have $\Pr[\mathbf{Y} = q] \leq 2^{-t}$. So, if we parse each q as $(q_1, q_2, \ldots, q_r) \in (\{0,1\}^\ell)^r$, we have that $\Pr[\mathbf{Y} = q] = \Pr[\mathbf{Y}_1 = q_1] \cdot \Pr[\mathbf{Y}_2 = q_2] \cdots \Pr[\mathbf{Y}_r = q_r]$ by the independence of these random variables, and so there must be some $\pi(q) \in [r]$ such that $\Pr[\mathbf{Y}_{\pi(q)} = q_{\pi(q)}] \leq 2^{-t/r}$.

Now, for any $x = (x_1, x_2, \ldots, x_r) \in (\{0,1\}^\ell)^r$, we let $x_{-i} := (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_r)$ denote x with its i^{th} chunk removed, and proceed as follows:

$$\begin{aligned}
\Pr[\mathbf{Y} \in Q - \mathsf{Bad}^*] &= \sum_{q \in Q - \mathsf{Bad}^*} \Pr[\mathbf{Y} = q] \\
&= \sum_{q \in Q - \mathsf{Bad}^*} \Pr[\mathbf{Y}_{\pi(q)} = q_{\pi(q)}] \cdot \Pr[\mathbf{Y}_{-\pi(q)} = q_{-\pi(q)}] \\
&\le 2^{-t/r} \sum_{q \in Q - \mathsf{Bad}^*} \Pr[\mathbf{Y}_{-\pi(q)} = q_{-\pi(q)}] \\
&\le 2^{-t/r} \sum_{q \in Q - \mathsf{Bad}^*} \Pr[\mathbf{Y} \in \mathsf{Ball}(q, \ell)] \\
&\le 2^{-t/r} \sum_{v \in \{0,1\}^n} \Pr[\mathbf{Y} = v] \cdot |\mathsf{Ball}(v, \ell) \cap Q| \\
&\le 2^{-t/r} \cdot \max_v |\mathsf{Ball}(v, \ell) \cap Q| \\
&\le 2^{-t/r} \cdot L \text{ if } \ell \le \rho n,
\end{aligned}$$

where the last line follows since Q is a (ρ, L)-list decodable code (see Section 3). Thus, provided that we have selected $r, \ell \in \mathbb{N}$ such that $r\ell = n$ and $\ell \le \rho n$, we obtain

$$\begin{aligned}
|\mathbf{X} - \mathbf{Q}| &\ge 1 - \Pr[\mathbf{Q} \in \mathsf{Bad}] - \max_W \Pr[\mathbf{X}^{(W)} \in Q - \mathsf{Bad}^{(W)}] \\
&> 1 - 2^{-k+t+r\log w} - 2^{-t/r+\log L}.
\end{aligned}$$

Before picking r, ℓ, we set $t = \frac{r}{r+1} \cdot (k - r\log w + \log L)$ as the value that equalizes the two exponents to $-\frac{1}{r+1} \cdot (k - r\log w + \log L) + \log L \le -\frac{k}{r+1} + \log(wL)$ to obtain

$$|\mathbf{X} - \mathbf{Q}| > 1 - 2^{-t/r+\log L+1} \ge 1 - 2^{-\frac{k}{r+1}+\log(wL)+1}. \tag{3}$$

Thus all that remains is to pick $r, \ell \in \mathbb{N}$ such that $r\ell = n$. If $1/\rho$ and ρn are integers, we simply set $r = 1/\rho$ and $\ell = \rho n$ and lower bound Equation (3) by $1 - 2wL \cdot 2^{-\frac{\rho}{1+\rho}k}$, as desired. If $1/\rho$ and ρn are not integers, it is easy to slightly modify the proof at the expense of a minor loss in parameters, as seen in the theorem statement: we refer the reader to the full version for more details. ◀

By combining Theorem 19 with the list decodability of (n, k, d) codes (Fact 1), we immediately obtain the following sampling lower bounds against (n, k, d) codes.

▶ **Theorem 20.** *Let $\mathbf{Q} \sim \{0,1\}^n$ be uniform over an (n, k, d) code. Then for any KRVZ sampler $\mathbf{X} \sim \{0,1\}^n$ of width w,*

$$|\mathbf{X} - \mathbf{Q}| \ge 1 - 8w \cdot 2^{-\frac{kd}{4n}}.$$

In the full version of the paper, we show that Theorem 20 is almost tight:

▶ **Theorem 21.** *There is a universal constant $C > 0$ such that the following holds. For all $n, k, d \in \mathbb{N}$ such that there exists a linear $[n, k, d]$ code, there exists a distribution $\mathbf{Q} \sim \{0,1\}^n$ uniform over a linear $[n, k, d]$ code that can be exactly generated by a (2^{-n})-granular KRVZ sampler $\mathbf{X} \sim \{0,1\}^n$ of width $w \le C \cdot 2^{C \cdot \frac{kd}{n} \cdot \log n}$.*

Finally, by combining Theorem 20 with the first bullet of the Equivalence Theorem 13, we immediately get our sampling lower bounds against codes for ROBPs (Theorem 1). Furthermore, by combining Theorem 21 with Lemma 15, we immediately get Remark 2.

Applications. Our sampling lower bounds against codes yield new data structure lower bounds against storing codewords succinctly and retrieving them using ROBPs (Corollary 4). This result follows immediately from a known connection, by Viola [22], between sampling lower bounds and data structure lower bounds. For a proof (and exposition of Viola's connection), we refer the reader to the full version of this paper. There, we also record generalizations of Corollary 4 to (n, k, d) codes and list decodable codes.

As a second application, we show in the full version of the paper that ROBPs of exponential width cannot test membership of a good code. In fact, we obtain a stronger result which shows that ROBPs of exponential width have exponentially small covariance with indicator functions of good codes.

6 A direct product theorem

In this section, we prove our direct product theorems. We start by showing a general direct product theorem for distributions, which we then specialize to get direct product theorems for KRVZ and ROBP samplers. To the best of our knowledge, these are the first direct product theorems for the task of sampling, and we hope that our general theorem will serve as a useful tool for extending these results to other computational models in future work.

A general direct product theorem. We now begin with our general direct product theorem for distributions. Intuitively, it says that if some computational class $\mathcal{C}$ has a hard time sampling a specific distribution $\mathbf{Q}$, and furthermore the distributions generated by that class are (almost) closed under a certain "slice property," then that same computational class has an extremely hard time sampling t independent copies of $\mathbf{Q}$.

▶ **Theorem 22.** *Let $\mathcal{X}$ be a family of distributions of the form $\mathbf{X} \sim \{0,1\}^n$, and let $\mathbf{Q} \sim \{0,1\}^n$ be a distribution such that $|\mathbf{X} - \mathbf{Q}| \geq \delta$ for every $\mathbf{X}$ in $\mathcal{X}$. Now, let $\mathcal{X}^*$ be a family of distributions of the form $\mathbf{X}^* = (\mathbf{X}_1^*, \ldots, \mathbf{X}_t^*) \sim (\{0,1\}^n)^t$, and suppose that $\mathcal{X}, \mathcal{X}^*$ have the $(\delta/2)$-*slice property*, meaning that for any $\mathbf{X}^* = (\mathbf{X}_1^*, \ldots, \mathbf{X}_t^*)$ in $\mathcal{X}^*$ and $i \in [t], x \in (\{0,1\}^n)^{i-1}$, it holds that $(\mathbf{X}_i^* \mid \mathbf{X}_{<i}^* = x)$ is $(\delta/2)$-close to some distribution in $\mathcal{X}$. Then for every $\mathbf{X}^* \in \mathcal{X}^*$, we have*

$$|\mathbf{X}^* - \mathbf{Q}^{\otimes t}| \geq 1 - e^{-t\delta^2/8}.$$

Consider now a computational class $\mathcal{C}$, and let $\mathcal{X}$ be the family of n-bit distributions that can be generated by $\mathcal{C}$, and let $\mathcal{X}^*$ be the family of nt-bit distributions that can be generated by $\mathcal{C}$. Notice that if $\mathcal{X}, \mathcal{X}^*$ have the slice property, then Theorem 22 implies a direct product theorem for sampling with $\mathcal{C}$. In what follows, we will prove Theorem 22, and show that KRVZ samplers and ROBP samplers have the slice property, thereby proving our direct product theorems for sampling in small space. In future work, it would be interesting to see if AC^0 circuits have the slice property, as this would immediately imply a direct product theorem for AC^0 via Theorem 22.

A simple new lemma on amplifying statistical distance. In order to prove Theorem 22, the key ingredient is the following simple new lemma, which may be of independent interest.

▶ **Lemma 23.** *Let $\mathbf{X} \sim V^n$ and $\mathbf{Y} \sim V^n$ each be a sequence of n random variables over V, where elements in the sequence need not be independent. Suppose that for all $i \in [n], v \in V^{i-1}$, we have $|(\mathbf{X}_i \mid \mathbf{X}_{<i} = v) - (\mathbf{Y}_i \mid \mathbf{Y}_{<i} = v)| \geq \delta$. Then it holds that $|\mathbf{X} - \mathbf{Y}| \geq 1 - e^{-n\delta^2/2}$.*

It turns out that this lemma will be a little cumbersome to prove using statistical distance. Thus, we will convert the statement into one about a more amenable notion of distance, known as the *Bhattacharyya coefficient*. Given two random variables $\mathbf{X}, \mathbf{Y}$ over some discrete space V, the Bhattacharyya coefficient between $\mathbf{X}$ and $\mathbf{Y}$ is denoted $\mathsf{BC}(\mathbf{X}, \mathbf{Y})$ and defined as $\mathsf{BC}(\mathbf{X}, \mathbf{Y}) := \sum_{v \in V} \sqrt{\Pr[\mathbf{X} = v] \cdot \Pr[\mathbf{Y} = v]}$. It is well-known and straightforward to show (using Cauchy-Schwarz) the following estimates on statistical distance in terms of the Bhattacharyya coefficient.

▶ **Fact 2.** *For any discrete random variables $\mathbf{X}, \mathbf{Y} \sim V$,*

$$1 - \mathsf{BC}(\mathbf{X}, \mathbf{Y}) \leq |\mathbf{X} - \mathbf{Y}| \leq \sqrt{2}\sqrt{1 - \mathsf{BC}(\mathbf{X}, \mathbf{Y})}.$$

We now prove a version of Lemma 23 that uses the Bhattacharyya coefficient, and then use the estimates from Fact 2 to obtain Lemma 23.

▶ **Lemma 24.** *Let $\mathbf{X} \sim V^n$ and $\mathbf{Y} \sim V^n$ each be a sequence of n random variables over V, where elements in the sequence need not be independent. Suppose that for all $i \in [n], v \in V^{i-1}$, we have $\mathsf{BC}((\mathbf{X}_i \mid \mathbf{X}_{<i} = v), (\mathbf{Y}_i \mid \mathbf{Y}_{<i} = v)) \leq \delta$. Then it holds that $\mathsf{BC}(\mathbf{X}, \mathbf{Y}) \leq \delta^n$.*

Proof. It is not too difficult to show, by induction, the slightly stronger statement that for any $i \in [n]$ it holds that $\mathsf{BC}(\mathbf{X}_{\leq i}, \mathbf{Y}_{\leq i}) \leq \delta^i$. We refer the reader to the full version for a complete proof. ◀

It is now straightforward to combine the above lemma with our estimates from Fact 2 to obtain Lemma 23. Given this simple new lemma on amplifying statistical distance, we return to proving Theorem 22.

Proof of Theorem 22. By the $(\delta/2)$-slice property and the triangle inequality, we know that each $|(\mathbf{X}^*_i \mid \mathbf{X}^*_{<i} = x) - \mathbf{Q}| \geq \delta/2$. The result now follows immediately from Lemma 23. ◀

A direct product theorem for KRVZ samplers. Given our general direct product theorem, it is now easy to prove the following direct product theorem for KRVZ samplers.

▶ **Theorem 25.** *Let $\mathbf{Q} \sim \{0,1\}^n$ be a distribution such that for any KRVZ sampler $\mathbf{X} \sim \{0,1\}^n$ of width w, it holds that $|\mathbf{X} - \mathbf{Q}| \geq \delta$. Then for any KRVZ sampler $\mathbf{X}^* \sim \{0,1\}^{nt}$ of width w,*

$$|\mathbf{X}^* - \mathbf{Q}^{\otimes t}| \geq 1 - e^{-t\delta^2/8}.$$

In order to prove this result, we start with the following fact, which can be easily verified using the definition of KRVZ samplers.

▶ **Fact 3.** *Let $\mathbf{X} \sim \{0,1\}^n$ be a KRVZ sampler of width w. Then for any $1 \leq i \leq j \leq n$ and any $x \in \{0,1\}^{i-1}$, the distribution $(\mathbf{X}_{i \to j} \mid \mathbf{X}_{<i} = x)$ is also a KRVZ sampler of width w.*

Notice that if $\mathcal{X}^*$ is the family of distributions $\mathbf{X}^* \sim (\{0,1\}^n)^t$ generated by KRVZ samplers of width w, and $\mathcal{X}$ is the family of distributions $\mathbf{X} \sim \{0,1\}^n$ of distributions generated by KRVZ samplers of width w, then Fact 3 asserts that $\mathcal{X}, \mathcal{X}^*$ have the 0-slice property. Thus by combining Theorem 22 with Fact 3, we immediately get Theorem 25.

A direct product theorem for sampling with ROBPs. At last, we are ready to prove the second main result of the paper, Theorem 5.

▶ **Theorem 26** (Theorem 5, restated). *Let $\mathbf{Q} \sim \{0,1\}^n$ be a distribution such that for any $\ell \in \mathbb{N}$ and any ROBP $F : \{0,1\}^\ell \to \{0,1\}^n$ of width w, it holds that $|F(\mathbf{U}_\ell) - \mathbf{Q}| \geq \delta$. Then for any $t, \ell^* \in \mathbb{N}$ and any ROBP $F^* : \{0,1\}^{\ell^*} \to \{0,1\}^{nt}$ of width w, it holds that*

$$|F^*(\mathbf{U}_{\ell^*}) - \mathbf{Q}^{\otimes t}| \geq 1 - e^{-t\delta^2/8}.$$

In order to prove this result, we would like to simply combine our Equivalence Theorem 13 with Theorem 25. However, since each direction of our equivalence theorem increases the width by a constant factor, this technique would only yield a *weak direct product theorem*: that is, the stronger statistical distance lower bounds of $1 - e^{-t\delta^2/8}$ would only hold for ROBPs of slightly smaller width (roughly $w/14$). We would like to avoid this, and keep the direct product theorem *strong*, in the sense that the width need not decrease at all.

Towards this end, the goal will be to prove a version of Fact 3 for sampling using ROBPs. That is, we would like to show that distributions generated by ROBPs have the *slice property*, as this can be combined with Theorem 22 to immediately obtain Theorem 26. In the following lemma, we prove that this is, indeed, the case.

▶ **Lemma 27.** *Let $F^* : \{0,1\}^{\ell^*} \to \{0,1\}^n$ be an ROBP of width w, and define $\mathbf{X} := F^*(\mathbf{U}_{\ell^*})$. For any $1 \leq i \leq j \leq n$ and $x \in \{0,1\}^{i-1}$ and $\varepsilon > 0$, there exists an ROBP $F : \{0,1\}^\ell \to \{0,1\}^{j-i+1}$ of width w and length $\ell = \ell^* + 3w\log(w/\varepsilon)$ such that*

$$|F(\mathbf{U}_\ell) - (\mathbf{X}_{i\to j} \mid \mathbf{X}_{<i} = x)| \leq \varepsilon.$$

The proof of Lemma 27 is significantly more challenging than the proof of Fact 3, and it is the crux of the proof to our second main theorem, Theorem 26. The key ingredient is a width 2 ROBP that can (approximately) simulate a coin with arbitrary bias, which we use to build a width w ROBP that can (approximately) generate an arbitrary distribution over $[w]$. Given such an object, it takes just a little more work to construct the ROBP from Lemma 27. For more details, we refer the reader to Section 2.2 and the full version of the paper.

At last, notice that if we set $\mathcal{X}^*, \mathcal{X}$ to be the set of all distributions of length nt and n (respectively) generated by ROBPs of width w, and set $\varepsilon := \delta/2$, then Lemma 27 implies that $\mathcal{X}, \mathcal{X}^*$ have the $(\delta/2)$-slice property. Thus by combining Theorem 22 with Lemma 27, we immediately get Theorem 26.

7 Future directions

The study of the complexity of sampling is still a very new area, and many open questions remain. For sampling with limited memory, it is natural to ask whether our results can be extended to more general models, such as read-k branching programs, or branching programs whose output bits need not be layered (as in Definition 10).[2] It would also be interesting to establish a separation between sampling with ROBPs and AC^0 circuits, in the direction not shown in Section 4. Namely, can one find a distribution that cannot be sampled in AC^0 but can be sampled by ROBPs? This is possible if one can construct an extractor (or even disperser) for AC^0 sources (distributions generated in AC^0), which can be computed by small width ROBPs. Since all known extractors for AC^0 sources [24] also work for distributions that can be generated in small space [10], new extractors are needed.

[2] For the latter model to be interesting, one can show that the number ℓ of random bits must be bounded, or else these objects can come arbitrarily close to sampling any distribution using just width 3.

References

1 Scott Aaronson. The equivalence of sampling and searching. *Theory of Computing Systems*, 55(2):281–298, 2014.

2 Andris Ambainis, Leonard J. Schulman, Amnon Ta-Shma, Umesh Vazirani, and Avi Wigderson. The quantum communication complexity of sampling. *SIAM Journal on Computing*, 32(6):1570–1585, 2003.

3 László Babai. Random oracles separate pspace from the polynomial-time hierarchy. *Information Processing Letters*, 26(1):51–53, 1987.

4 Paul Beame. A general sequential time-space tradeoff for finding unique elements. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 197–203, 1989.

5 Chris Beck, Russell Impagliazzo, and Shachar Lovett. Large deviation bounds for decision trees and sampling lower bounds for AC0-circuits. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 101–110. IEEE, 2012.

6 Itai Benjamini, Gil Cohen, and Igor Shinkar. Bi-Lipschitz bijection between the Boolean cube and the Hamming ball. *Israel Journal of Mathematics*, 212(2):677–703, 2016.

7 Ravi B. Boppana and Jeffrey C. Lagarias. One-way functions and circuit complexity. *Information and Computation*, 74(3):226–240, 1987.

8 Allan Borodin and Stephen Cook. A time-space tradeoff for sorting on a general sequential model of computation. *SIAM Journal on Computing*, 11(2):287–297, 1982.

9 Allan Borodin, Michael J. Fischer, David G. Kirkpatrick, Nancy A. Lynch, and Martin Tompa. A time-space tradeoff for sorting on non-oblivious machines. In *20th Annual Symposium on Foundations of Computer Science (sfcs 1979)*, pages 319–327. IEEE, 1979.

10 Eshan Chattopadhyay and Jesse Goodman. Improved extractors for small-space sources. To appear in the 62nd Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2021.

11 Eshan Chattopadhyay, Jesse Goodman, and David Zuckerman. The space complexity of sampling. *Electron. Colloquium Comput. Complex.*, page 106, 2021. URL: `https://eccc.weizmann.ac.il/report/2021/106`.

12 Anindya De and Thomas Watson. Extractors and lower bounds for locally samplable sources. *ACM Transactions on Computation Theory (TOCT)*, 4(1):1–21, 2012.

13 Oded Goldreich, Shafi Goldwasser, and Asaf Nussboim. On the implementation of huge random objects. *SIAM Journal on Computing*, 39(7):2761–2822, 2010.

14 Mika Göös and Thomas Watson. A lower bound for sampling disjoint sets. *ACM Transactions on Computation Theory (TOCT)*, 12(3):1–13, 2020.

15 Johan Håstad. *Computational limitations of small-depth circuits*. MIT press, 1987.

16 Russell Impagliazzo and Moni Naor. Efficient cryptographic schemes provably as secure as subset sum. *Journal of cryptology*, 9(4):199–216, 1996.

17 Rahul Jain, Yaoyun Shi, Zhaohui Wei, and Shengyu Zhang. Efficient protocols for generating bipartite classical distributions and quantum states. *IEEE Transactions on Information Theory*, 59(8):5171–5178, 2013.

18 Jesse Kamp, Anup Rao, Salil Vadhan, and David Zuckerman. Deterministic extractors for small-space sources. *Journal of Computer and System Sciences*, 77(1):191–220, 2011.

19 Robert Koenig and Ueli Maurer. Extracting randomness from generalized symbol-fixing and markov sources. In *International Symposium on Information Theory, 2004. ISIT 2004. Proceedings.*, page 232. IEEE, 2004.

20 Robert Koenig and Ueli Maurer. Generalized strong extractors and deterministic privacy amplification. In *IMA International Conference on Cryptography and Coding*, pages 322–339. Springer, 2005.

21 Shachar Lovett and Emanuele Viola. Bounded-depth circuits cannot sample good codes. *Comput. Complex.*, 21(2):245–266, 2012. `doi:10.1007/s00037-012-0039-3`.

22 Emanuele Viola. The complexity of distributions. *SIAM Journal on Computing*, 41(1):191–218, 2012.

23 Emanuele Viola. Extractors for Turing-machine sources. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 663–671. Springer, 2012.

24 Emanuele Viola. Extractors for circuit sources. *SIAM Journal on Computing*, 43(2):655–672, 2014.

25 Emanuele Viola. Quadratic maps are hard to sample. *ACM Transactions on Computation Theory (TOCT)*, 8(4):1–4, 2016.

26 Emanuele Viola. Sampling lower bounds: boolean average-case and permutations. *SIAM Journal on Computing*, 49(1):119–137, 2020.

27 Thomas Watson. Time hierarchies for sampling distributions. *SIAM Journal on Computing*, 43(5):1709–1727, 2014.

28 Thomas Watson. Nonnegative rank vs. binary rank. *Chicago Journal of Theoretical Computer Science*, 2016(2), February 2016.

29 Thomas Watson. Communication complexity with small advantage. *computational complexity*, 29(1):1–37, 2020.

30 Andrew C. Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*, pages 80–91. IEEE, 1982.