Proofs of Proximity for Distribution Testing*†

Alessandro Chiesa[1] and Tom Gur[2]

1 **UC Berkeley, California, USA**
`alexch@berkeley.edu`
2 **UC Berkeley, California, USA**
`tom.gur@berkeley.edu`

Abstract

Distribution testing is an area of property testing that studies algorithms that receive few samples from a probability distribution $\mathcal{D}$ and decide whether $\mathcal{D}$ has a certain property or is far (in total variation distance) from all distributions with that property. Most natural properties of distributions, however, require a large number of samples to test, which motivates the question of whether there are natural settings wherein fewer samples suffice.

We initiate a study of proofs of proximity for properties of distributions. In their basic form, these proof systems consist of a tester (or verifier) that not only has sample access to a distribution but also explicit access to a proof string that depends arbitrarily on the distribution. We refer to these as NP distribution testers, or MA distribution testers if the tester is a probabilistic algorithm. We also study IP distribution testers, a more general notion where the tester interacts with an all-powerful untrusted prover.

We investigate the power and limitations of proofs of proximity for distributions and chart a landscape that, surprisingly, is significantly different from that of proofs of proximity for functions. Our main results include showing that MA distribution testers can be quadratically stronger than standard distribution testers, but no stronger than that; in contrast, IP distribution testers can be exponentially stronger than standard distribution testers, but when restricted to public coins they can be quadratically stronger at best.

1998 ACM Subject Classification F.1.2 [Modes of Computation]: Probabilistic computation

Keywords and phrases distribution testing, proofs of proximity, property testing

Digital Object Identifier 10.4230/LIPIcs.ITCS.2018.53

1 Introduction

Distribution testing, introduced by Goldreich and Ron [34] and Batu et al. [8], is an area of property testing [55, 30] that studies sublinear-time algorithms for approximate decision problems regarding probability distributions over massive domains. Such algorithms, known as *distribution testers*, are given independent samples from an unknown distribution and are required to decide whether the distribution has a certain property, or is far from having it. More precisely, a distribution tester for a property Π of distributions over a domain Ω is a probabilistic algorithm that, given a proximity parameter $\varepsilon > 0$, determines whether a distribution $\mathcal{D}$ over Ω has the property Π or is ε-far (typically, in total variation distance) from any distribution that has Π, by drawing a sublinear number of independent samples from $\mathcal{D}$.

* This work was supported in part by the UC Berkeley Center for Long-Term Cybersecurity.
† A full version of the paper is available at [23], `https://eccc.weizmann.ac.il/report/2017/155`

In the last two decades distribution testing has received much attention, not only because it asks fundamental questions about distributions but also because it has applications ranging from statistical hypothesis testing [43] and model selection [14] to property testing [34, 17] and biology [62, 48]. A long line of works, including [7, 46, 6, 47, 49, 58, 3, 12, 44, 22, 11, 24, 57], has investigated many natural properties of distributions, determining the sample complexity of core problems such as testing uniformity, support size, identity to a specified distribution, and many more (see recent surveys [53, 16] and a forthcoming book [29]).

Whereas testing properties of functions is often possible with few queries (independently of the function's domain size), testing properties of distributions typically requires many samples. In particular, the vast majority of properties of distributions studied in the literature require $\Omega(\sqrt{n})$ samples to test, where n is the domain size. This state of affairs has motivated researchers to study distribution testing using stronger types of access to the distribution [18, 26, 2, 21], in which the tester can draw samples conditioned on a subset of the domain, and models in which the tester is granted additional access to the cumulative distribution function or probability mass function of the distribution [54, 15]. In this work we take a different approach: we allow the tester to be aided by a prover, but keep the standard sample access to the distribution (without any conditioning), as we now explain.

A fundamental question that arises in any computational model is to understand the power of a 'proof'. Indeed, the famous $\mathbf{P} \neq \mathbf{NP}$ conjecture, which is concerned with the power of proofs in the setting of polynomial-time computation, is widely considered as one of the most important open problems in the theory of computation. Moreover, proof systems are studied in many other settings, such as communication complexity [4, 1, 42], quantum computation [61, 50, 59], data streams [19, 38, 20], and, most relevant to this work, property testing, as we now recall.

Proofs in the *functional* (standard) setting of property testing are known as *proofs of proximity* [25, 9]. These are probabilistic proof systems in which the verifier makes a sublinear number of queries to a statement, and is only required to reject statements that are far from true. In a Merlin–Arthur proof of proximity (MAP) [39], the verifier receives explicit access to a proof of sublinear length, in addition to query access to the statement. More generally, in an interactive proof of proximity (IPP) [52], the verifier interacts with an all-powerful untrusted prover. MAPs and IPPs have been studied in a line of recent works, including [27, 33, 41, 32, 28, 31, 51, 40, 10], and may be thought of as the MA (i.e., "randomized NP") and IP analogues of functional property testing, respectively.

In this work, we initiate a study of proof systems for testing properties of *distributions*, i.e., proofs of proximity for distribution testing. We define several natural types of proofs, and investigate their power and limitations. The landscape that we chart turns out to be completely different, both qualitatively and quantitatively, from that for proofs of proximity for *functions*. We now discuss our results, first on non-interactive proofs and then on interactive proofs.

2 Non-interactive proofs of proximity for distribution testing

We study a natural analogue of the notion of NP proofs for testing properties of distributions. Letting $\Delta(\Omega_n)$ be the set of distributions over a domain Ω, and letting $\Pi \subseteq \Delta(\Omega)$ be a property, the tester is given sample access to a distribution $\mathcal{D} \in \Delta(\Omega)$ and explicit access to a proof π and proximity parameter ε. We require that for every distribution $\mathcal{D} \in \Pi$ there exists a proof π such that the tester accepts, and for every distribution $\mathcal{D}$ that is ε-far from Π and every proof the tester rejects, both with high probability (e.g., 2/3).

Following standard conventions, if such a tester is a *deterministic* algorithm (i.e., is not allowed to toss coins), then we call it an NP distribution tester, and if it is a *probabilistic* algorithm, then we call it an MA distribution tester. As we discuss later, in stark contrast to proofs of proximity for *functions*, for which deterministic testers are degenerate [38], the power of MA distribution testers and NP distribution testers is essentially equivalent. Thus we henceforth present our results for MA distribution testers only, and remark that these results qualitatively translate to NP distribution testers as well.

Analogously to prior work in distribution testing and proximity proofs, we consider two main efficiency measures for MA distribution testers: (a) *sample complexity*, which is the number of samples drawn by the tester from the distribution; (b) *proof complexity*, which is the length of the honest proof. Both complexity measures are functions of the domain size and the proximity parameter.

Perhaps the first question that arises in this direction is whether verification can be cheaper than decision. In other words, are MA distribution testers stronger than standard distribution testers? For functional proofs of proximity the answer is immediate: every property can be tested with just $O(1)$ queries to the input, when given a linear-size proof. This proof simply contains a description of the input, in which case the tester can read the entire proof, decide membership in the property, and query the input at few random locations to check that it is close to the proof. Linear-size proofs thus trivialize testing properties of functions.

In distribution testing, however, the situation is not as simple. For starters, given a purported description of $\mathcal{D}$, checking that this description actually matches the input distribution typically requires more than a constant number of samples. Moreover, the description of a distribution $\mathcal{D}$ may be very large (even infinite), and so the proof cannot simply contain its description. Nonetheless, these difficulties can be dealt with, albeit at the cost of higher complexity.

To simplify exposition, throughout the introduction we fix a domain Ω_{n} of size n and fix the proximity parameter ε to a small constant. Our first result shows that proofs of (nearly) linear length allow testing *any* property with only $O(\sqrt{\mathsf{n}})$ samples; moreover, there are natural properties for which the sample complexity can be smoothly reduced (down to constant) using increasingly longer proofs.

▶ **Theorem 1** (informal; see full version for details)**.**

1. *For any property* $\Pi \subseteq \Delta(\Omega_{\mathsf{n}})$*, there exists an MA distribution tester with proof complexity* $O(\mathsf{n}\log(\mathsf{n}))$ *and sample complexity* $\mathsf{s} = O(\max_{\mathcal{D}\in\Pi} \|D\|_{2/3}) = O(\sqrt{\mathsf{n}})$*. (Here* $\|\cdot\|_{2/3}$ *is the* $\ell_{2/3}$ *quasi-norm.)*
2. *There exists a (natural) property* $\Pi \subseteq \Delta(\Omega_{\mathsf{n}})$ *for which every distribution tester uses* $\tilde{\Omega}(\mathsf{n})$ *samples, yet there is an MA distribution tester for* Π *with proof complexity* $O(\mathsf{n}\log(\mathsf{n}))$ *and sample complexity* $O(1)$*. Furthermore, one can trade proof against sample complexity and, e.g., make both complexities* $\tilde{O}(\sqrt{\mathsf{n}})$*.*

We remark that the second item of Theorem 1 is proved with respect to a *promise* problem.

Theorem 1 confirms the intuition that MA distribution testers are stronger than standard distribution testers. However, while in the settings of proximity proofs for functions it is possible to obtain exponential savings in query complexity, even using proofs of merely logarithmic length [39], our Theorem 1 only shows MA distribution testers in which the product of the proof and sample complexities is at least as large as the sample complexity

of standard distribution testers.[1] This discussion raises the question of whether there exist stronger MA distribution testers, or whether non-interactive proofs of proximity for distributions are indeed more limited than their functional counterparts.

Furthermore, Theorem 1 shows that the sample complexity of MA distribution testers for any property can be reduced to $O(\sqrt{n})$. Yet, for properties that can be tested (without a proof) using $O(\sqrt{n})$ samples, is it always the case that MA distribution testers can be stronger than standard distribution testers?

To answer the questions above, we study the *limitations* of non-interactive proofs of proximity for distributions. Our next result shows that for *every* property and every MA distribution tester, either its proof or its sample complexity can at best be quadratically better than the (optimal) sample complexity of a standard distribution tester. Moreover, there also exists a natural property (the property of being uniformly distributed) for which MA distribution testers cannot do better than standard distribution testers.

▶ **Theorem 2** (informal; see full version for details)**.** *Let* s_Π *be the optimal sample complexity for testing a property* Π *without the aid of any proofs.*

- *For every* $\Pi \subseteq \Delta(\Omega_{\mathsf{n}})$ *and every MA distribution tester for* Π *with proof complexity* p *and sample complexity* s*, it holds that* $\mathsf{p} \cdot \mathsf{s} = \Omega(\mathsf{s}_\Pi)$*.*
- *Every MA distribution tester for the uniformity property* U_{n} *has sample complexity* $\Omega(\mathsf{s}_{\mathsf{U}_{\mathsf{n}}}) = \Omega(\sqrt{\mathsf{n}})$*, regardless of its proof complexity.*

Theorem 2 thus shows that the upper bounds in Theorem 1 are tight, up to logarithmic factors. (The first item of Theorem 2 shows the tightness of the second item of Theorem 1, and the second item of Theorem 2 shows the tightness of the first item of Theorem 1 with respect to a particular property.)

On derandomizing MA distribution testers.

As mentioned above, the power of deterministic verification (NP proofs) and randomized verification (MA proofs) is essentially equivalent in the setting of distribution testing. More accurately, the following theorem shows that MA distribution testers can be derandomized into NP distribution testers at the price of only a small increase in sample complexity.

▶ **Theorem 3** (informal; see full version for details)**.** *Every MA distribution tester with proof complexity* p *and sample complexity* s *can be emulated by an NP distribution tester with proof complexity* p *and sample complexity* $O\big(\mathsf{s} + \log(\mathsf{n})\big)$*.*

We remark that a direct proof for the special case of standard testers (without access to a proof) is sketched in [29, Chapter 11].

3 Interactive proofs of proximity for distribution testing

While MA distribution testers are stronger than standard distribution testers, they are limited to multiplicatively trading off sample complexity for proof complexity. Can one do even better with other types of proof systems? To study this question, we consider a natural analogue of *interactive proofs* [36] in the setting of distribution testing.

[1] To see this holds with respect to the first item of Theorem 1, recall that every property can be tested using $O(\mathsf{n})$ samples (for a constant value of the proximity parameter).

An *IP distribution tester* generalizes the notion of an MA distribution tester by allowing the tester to interact with an all-powerful untrusted prover who knows everything about the input distribution $\mathcal{D}$. The prover tries to convince the tester that $\mathcal{D}$ has a certain property Π. If $\mathcal{D} \in \Pi$ then there exists a prover strategy that makes the tester accept with high probability; if instead $\mathcal{D}$ is far from Π then the tester rejects with high probability regardless of prover strategy.

Similarly to the non-interactive setting, we seek to minimize the sample complexity, as well as *communication complexity*, which is the total number of bits exchanged between the two parties (and generalizes proof complexity). We also consider the *round complexity*, which is the number of rounds of interaction, where each round consists of a message from one party to the other and its reply.

The next theorem shows that it is possible to test properties of distributions much more efficiently by interacting with a prover than by receiving a non-interactive proof. In fact, even a single round of interaction suffices to obtain *exponential* savings in communication and sample complexity compared to the sample complexity of standard distribution testers (and hence MA distribution testers as well).

▶ **Theorem 4** (informal, see full version). *There exists a property $\Pi \subseteq \Delta(\Omega_{\mathsf{n}})$ such that:*

1. *there is a 1-round IP distribution tester for Π with communication complexity $O(\log(\mathsf{n}))$ and sample complexity $O(1)$; yet*
2. *every (standard) distribution tester for Π must use $\tilde{\Omega}(\sqrt{\mathsf{n}})$ samples.*

A fundamental distinction between types of interactive proofs is according to how the tester uses its own randomness. The interaction is public-coin if the tester reveals the outcome of its coins immediately after tossing them; it is private-coin if the tester can keep such outcomes to itself. Public-coin interactive proofs are called AM proofs [5], and so we call their distribution testing analogues AM distribution testers. We stress that in these public-coin protocols, the prover does *not* see the samples drawn by the tester.

Goldwasser and Sipser [37] proved that the expressive power of private-coin interactive proofs is essentially equivalent to that of public-coin interactive proofs, despite the latter being syntactically weaker. Rothblum, Vadhan, and Wigderson [52] observed that [37]'s proof of this statement extends to the setting of interactive proofs of proximity for *functions*. The next theorem shows that, unlike in the aforementioned models, the power of public-coin interaction for testing distributions is rather limited, *regardless of round complexity*.

▶ **Theorem 5** (informal, see full version). *For every property $\Pi \subseteq \Delta(\Omega_{\mathsf{n}})$ and $\mathsf{r} \in \mathbb{N}$ (not necessarily a constant), it holds that every r-round AM distribution tester for Π with communication complexity c and sample complexity s satisfies $\mathsf{c} \cdot \mathsf{s} = \Omega(\mathsf{s}_\Pi)$. (As before, s_Π denotes the optimal sample complexity for testing property Π without the aid of any proofs.)*

We note that the combination of our Theorems 4 and 5 yields an *exponential* separation between the power of IP distribution testers and AM distribution testers, which stands in stark contrast to the equivalence of private-coin and public-coin interaction in the functional setting.

While their power is limited when compared to IP distribution testers, AM distribution testers are still stronger than standard distribution testers, and possibly MA distribution testers as well. In the full version we show an AM distribution tester for a natural property that tightly matches the lower bound in Theorem 5, and also allows for smooth communication versus sample complexity tradeoffs. It is an open problem whether this upper bound can also be obtained via MA distribution testers, or whether public coin interaction in the setting of distribution testing is strictly stronger.

Table 1 Comparison between proofs of proximity for testing distributions and testing functions.

		Testing Distributions this work	**Testing Functions** [52, 39, 27, 40]
non-interactive proofs	**Proofs of linear length**	reduce sample complexity of *any* property to $O(\sqrt{n})$	reduce sample complexity of *any* property to $O(1)$
	MA proofs of proximity vs. standard testers	quadratically stronger	exponentially stronger
	Probabilistic (MA) vs. deterministic (NP) verification	nearly equivalent	NP proofs of proximity are extremely weak
	Hardest property for non-interactive proofs	explicit and natural; no better than standard testers, regardless of proof length	non-explicit (random property); linear length proof is required to outperform standard testers
interactive proofs	**Private vs. public coin protocols**	exponential separation	almost equivalent
	AM round hierarchy coin protocols	AM complexity is quadratically related to the sample complexity of standard testers	there is a property for which the AM complexity is $\approx n^{1/r}$ for r-round protocols

4 Comparison of functional and distributional proofs of proximity

In this work we consider several fundamental questions about proofs of proximity that were previously studied for properties *of functions*. We study these questions for properties *of distributions* instead.

One may naively expect that, since we are asking similar questions, we should obtain similar answers. However our results demonstrate that proofs of proximity for distributions behave dramatically different, both qualitatively and quantitatively, from proofs of proximity for functions. We summarize these different "complexity landscapes" in Table 1.

In retrospect these dramatic differences are easily interpreted. First and foremost, even standard (function) property testing and distribution testing are dissimilar: not only the tested objects are structurally different, but, just as importantly, the *access* to these objects is different as well (query access versus sample access). Moreover, these differences are more pronounced with regard to proofs of proximity because proof techniques to reason about them are very sensitive to input representation and access type. This is indeed what we find when inspecting our proof techniques, and the reasons for why our results hold.

5 Techniques

We establish our results via an eclectic set of technical tools that varies from section to section. These include extraction and derandomization, reductions from SMP communication complexity, lifting lemmas, granular approximation, and tolerant testing. To facilitate understanding of the main ideas behind each result, in the technical sections we precede the formal proof of each result with an intuitive high-level overview.

Below, we provide a taste of our techniques, grouped according to whether they give us upper bounds (Section 5.1), lower bounds (Section 5.2), or derandomization (Section 5.3).

5.1 Upper bounds

We overview the techniques that we use to obtain: a generic upper bound for MA distribution testers (first item of Theorem 1), an improved MA upper bound for a particular property (second item of Theorem 1), and an IP distribution tester that is exponentially more efficient than any MA distribution tester (first item of Theorem 4).

A generic MA upper bound.

We sketch a proof of a special case of Theorem 1, showing that *any* property can be tested via an MA distribution tester that uses $O(\sqrt{n}/\varepsilon^2)$ samples and a proof of linear size. The idea is that a linear-size proof π can allegedly consist of a description of the input distribution $\mathcal{D} \in \Pi$. Since the tester has explicit access to π and our goal is to minimize *sample* complexity (and not *time* complexity), the MA distribution tester can directly check membership of π in the property Π, reducing the problem to testing that the input distribution $\mathcal{D}$ is identical to π, a task that can be performed via $O(\sqrt{\mathsf{n}}/\varepsilon^2)$ samples [56].

One problem that arises is that, unlike the setting of testing Boolean functions or graphs, in the setting of distribution testing the size of the description of $\mathcal{D}$ may be very large (even infinite). To overcome this, we let an honest proof consist of a *granular* approximation $\mathcal{D}'$ of $\mathcal{D}$, where the mass of each element in the support of $\mathcal{D}'$ is a multiple of $m := \Theta(1/\mathsf{n})$; this approximation has at most linear size.

Note, however, that it could be the case that $\mathcal{D} \in \Pi$, whereas its granular approximation $\mathcal{D}'$ is close to Π but not in Π (similarly, $\mathcal{D}$ may by ε-far from Π, whereas may $\mathcal{D}'$ not). Nevertheless, using a *tolerant* testing procedure, the tester can ensure that with high probability it would rule regarding $\mathcal{D}'$ just as it would regarding $\mathcal{D}$, and so the granular approximation suffices to this end.

MA distribution tester with sublinear proofs.

To simplify the following presentation, we restrict our attention to m-granular distributions over the domain $[\mathsf{n}]$, for some $m = \Omega(1/\mathsf{n})$.

Consider the *gap isolated elements* problem, which is the problem of deciding whether a distribution $\mathcal{D}$ has a large number of isolated elements, or only a small one, where an element $i \in [\mathsf{n}]$ is said to be isolated if $\mathcal{D}$ is not supported on its adjacent elements $i-1$ and $i+1$.

We sketch an MA distribution tester with proof and sample complexity $\tilde{O}(\sqrt{\mathsf{n}})$ that accepts distributions with at least $\sqrt{\mathsf{n}}$ isolated elements and rejects distributions with at most $\sqrt{\mathsf{n}}/2$. (In the full version we show proof versus sample complexity tradeoffs for a wide range of parameterizations of this problem.)

The proof string simply specifies $\sqrt{\mathsf{n}}$ allegedly isolated elements of the input distribution $\mathcal{D}$, and the MA distribution tester draws $O(\sqrt{\mathsf{n}})$ samples and accepts if and only if all of the samples are not adjacent to the elements specified by the prover. Of course, if $\mathcal{D}$ indeed has at least $\sqrt{\mathsf{n}}$ isolated elements, the proof can specify them, and the MA distribution tester will accept with probability 1.

The key point is that if $\mathcal{D}$ has at most $\sqrt{\mathsf{n}}/2$ isolated elements, then every purported proof must specify at least $\sqrt{\mathsf{n}}/2$ elements that have an adjacent element on which $\mathcal{D}$ is supported on. Denote these supported adjacent elements by B, and note that every element of B is in fact a local certificate that $\mathcal{D}$ is a no-instance; that is, if the tester draws a *single* element in B, it can safely reject. By the granularity of $\mathcal{D}$ the total mass of B is $\Omega(1/\sqrt{\mathsf{n}})$, and so it suffices to draw $O(\sqrt{\mathsf{n}})$ samples to hit B with high probability.

IP distribution tester with logarithmic complexity.

We sketch an IP distribution tester for the isolated elements problem that has logarithmic communication complexity and constant sample complexity. (In the full version we also show that any public-coin IP distribution tester, and in particular standard and MA distribution testers, has exponentially larger complexity.)

Here we use different parameter settings than above, and in fact we shall not need the gap (promise problem) variant, and simply consider the property

$$\Pi_{\mathsf{Isolated}} := \{\mathcal{D} \in \Delta([\mathsf{n}]) \mid \forall i \in [\mathsf{n}]\ i \notin \mathrm{supp}(D) \text{ or } (i+1) \notin \mathrm{supp}(D)\}\ ;$$

that is, all distributions (not necessarily granular) in which no two consecutive elements are supported.

Consider the following IP distribution tester for this property. The tester draws $O(1/\varepsilon)$ samples from the input distribution $\mathcal{D}$ and *masks* these samples by shifting each sample to its subsequent element with probability 1/2. The tester then sends the masked samples to the prover and asks the prover to recover the original samples (prior to the shifts).

The point is that if the supported elements of $\mathcal{D}$ are indeed isolated, then the prover can always determine the original samples (as $\mathcal{D}$ cannot be supported on both an element and its shift). On the other hand, if $\mathcal{D}$ is ε-far from Π_{Isolated}, then there exist adjacent supported elements whose weight is $\Omega(\varepsilon)$, an so the prover is forced to guess which samples were shifted and which not, and will get caught with constant probability.

5.2 Lower bounds

Our lower bounds are all based on the following paradigm: we first prove a lower bound on the complexity of BPP distribution testers, typically via a reduction from SMP communication complexity, and then use "lifting" lemmas that allow us to transfer this lower bound to MA and AM distribution testers (where recall that by the latter we refer to public-coin *interactive* proof systems). We illustrate this methodology by sketching a proof of lower bounds on the complexity of MA and AM distribution testers for the isolated elements property Π_{Isolated}, which consists of all distributions in which no two consecutive elements are supported.

BBP lower bound via reduction from communication complexity.

We use the SMP communication complexity method [13]. Recall that, in a private-coin SMP protocol for a predicate f, the players Alice and Bob are given strings $x, y \in \{0,1\}^k$ (respectively), and each of the players is allowed to send a message, which depends on the player's input and *private* randomness, to a referee who is then required to decide whether $f(x,y) = 1$ by only looking at the players' messages and flipping coins. It is well-known that for the equality predicate ($f(x,y) = 1 \leftrightarrow x = y$), every such protocol must communicate $\Omega(\sqrt{k})$ bits [45].

Let P contain each third element of the domain, i.e., $P := \{3j - 1 \mid j \in [(\mathsf{n}-1)/3]\}$. Our reduction will map (a) yes-instances of EQ_k to distributions that are uniform over $|P|$ isolated elements; and (b) no-instances of EQ_k to distributions wherein for an ε-fraction of $p \in P$ it holds that $\mathcal{D}(p) = \Omega(1/\mathsf{n})$ and $\mathcal{D}(p+1) = \Omega(1/\mathsf{n})$, hence D is ε-far from Π_{Isolated}. Details follow.

Assume there exists a tester for Π_{Isolated} with sample complexity s. Each of the players encodes its input string via a balanced asymptotically good code ECC (that is, ECC: $\{0,1\}^k \to \{0,1\}^n$ with constant rate and relative distance $\varepsilon = \Omega(1)$, such that each codeword of

ECC contains the same number of 0's and 1's). Alice and Bob each draw $O(s)$ samples that are uniformly distributed over P, and *shift* each sample according to $\mathrm{ECC}(x)$ and $\mathrm{ECC}(y)$, respectively. That is, Alice sends to the referee independent samples uniformly drawn from $A := \left\{ i + \mathrm{ECC}(x)_{(i+1)/3} \mid i \in P \right\}$, and Bob sends samples uniformly drawn from $B := \left\{ i + \mathrm{ECC}(y)_{(i+1)/3} \mid i \in P \right\}$. Finally, the referee invokes the tester for Π_{Isolated} with respect to the distribution $\frac{1}{2}\mathsf{U}_{\mathsf{n}}(A) + \frac{1}{2}\mathsf{U}_{\mathsf{n}}(B)$, emulating each draw by tossing a random coin and deciding accordingly whether to use a sample by Alice or Bob.

The point is that if $x = y$, then $\mathrm{ECC}(x) = \mathrm{ECC}(y)$, and so both players shift their samples (which are in P, and so separated by two non-supported elements) in the same way, and so the resulting mixed distribution is uniform over isolated elements. On the other hand, if $x \neq y$, then $\mathrm{ECC}(x)$ is ε-far from $\mathrm{ECC}(y)$, and so the resulting distribution will have roughly $\varepsilon \cdot |P|$ non-isolated elements of weight $\Omega(1/|P|)$ each. Thus, we have $s = \tilde{\Omega}(\sqrt{k}) = \tilde{\Omega}(\sqrt{n})$.

Lifting the BPP lower bound to MA and r-round AM distribution testers.

We begin with the simpler task of proving an MA lower bound on Π_{Isolated}. To lift the BPP lower bound we proved above to MA, we show that any MA distribution tester T for any property Π (in particular, Π_{Isolated}) with proof complexity p and sample complexity s can be emulated by a BPP distribution tester T' with sample complexity $O(\mathsf{p} \cdot \mathsf{s})$.

The key observation is that the samples that T draws are completely independent of the *proof* that it receives. Since we aim to minimize sample complexity (rather than time complexity), we can hope to emulate all possible proofs, while reusing the samples. However, since there are exponentially many (2^{p}) possible proofs, we need to amplify the soundness to assure no error occurs with high probability. To this end, at the cost of increasing the sample complexity to $O(\mathsf{p} \cdot \mathsf{s})$, we invoke the tester $O(\mathsf{p})$ times to obtain soundness error $\exp(-\mathsf{p})$, which suffices to take a union bound over invocations of the amplified T with respect to all possible proofs.

To lift the BPP lower bound to r-round AM distribution testers, for *any* (possibly non-constant) $r \geq 1$, we need a significantly more involved argument. Recall that an AM distribution tester works as follows. In each round, the tester samples fresh randomness ρ_i and sends it to the prover, which replies with a message m_i that may arbitrarily depend on the input distribution $\mathcal{D} \in \Delta(\Omega_{\mathsf{n}})$, proximity parameter ε, and transcript of the interaction so far. After receiving the last message from the prover, the tester draws samples from $\mathcal{D}$ and decides according to these samples, proximity parameter, and transcript of the entire interaction.

Analogously to the proof of the MA lifting lemma, the high-level idea is that since the samples drawn from $\mathcal{D}$ are independent of the transcript of interaction, a BPP distribution tester can emulate all possible interactions, while using the *same* samples for *all* invocations. However, several difficulties arise when trying to naively implement the foregoing idea.

First, note that the tester cannot simply emulate the optimal prover, because it is determined by a distribution from which it only has few samples. Second, we cannot afford to enumerate over all prover *strategies*, as there is a doubly exponential number of them (each strategy is a function from the space of previous transcripts to the next message). Instead, we can only afford enumerating over all possible *transcripts*, which are *not* uniformly generated. Third, as before, since we invoke the tester with respect to exponentially many transcripts, we need to reduce its soundness error accordingly. Unfortunately, amplifying the soundness would result in an increase in communication complexity, which we cannot afford. Finally, even given exponentially small soundness error, whereas for MA it suffices to find a single proof that is accepted with high probability, here there may exist specific

transcripts in which the prover fools the tester with probability 1 (this is because we consider transcripts, rather than prover strategies).

A key step towards overcoming these difficulties is to rely on a simple yet important observation: each AM distribution tester induces a family of BPP distribution testers that are determined by the interaction. That is, since the *transcript* of the interaction is a random variable that is *independent of the samples* drawn by the AM distribution tester, the interaction phase can be viewed as a procedure that defines a BPP distribution tester that is invoked after this phase. In particular, this allows us to perform soundness amplification *solely on the induced BPP distribution testers*.

The procedure above implies that, with high probability over the random messages of the tester, each of the corresponding induced BPP distribution testers decides correctly, with only an exponentially small probability of error, without incurring any blowup in communication complexity. (Note, however, that the total soundness of the AM distribution tester does not necessarily increase significantly.)

Thus, we can invoke all the BPP distribution testers that are induced by all possible transcripts, while reusing the same samples for all invocations, such that with high probability no error will occur in any of the relevant invocations. Finally, we show that the the interaction tree induced by these invocations is significantly different for yes-instance and no-instances, and so the tester can consider it and decide whether there exists a prover strategy that would have been accepted with high probability by the AM distribution tester.

5.3 Derandomization

The key observation behind the derandomization of MA distribution testers (Theorem 3) is that while an NP distribution tester is a *deterministic* algorithm, it receives *random* samples from the input distribution $\mathcal{D}$. Thus we can hope to simulate the coin tosses of the MA distribution tester by deterministically extracting the necessary randomness from the samples.

To deterministically extract uniform bits from independent samples drawn from a distribution $\mathcal{D} \in \Delta([\mathsf{n}])$, we arbitrarily group the samples into pairs, discard pairs in which both samples are the same, then write 1 (respectively, 0) for every pair in which the first element is larger (respectively, smaller) than the second. Since the samples are independent, the first sample of each pair is equally likely to be larger as it is to be smaller than the second sample, and so we obtain a uniformly distributed string. This procedure can be thought of as generalizing the seedless extractor of Von Neumann [60].

The foregoing approach raises two concerns: (a) if $\mathcal{D}$ has small entropy, each bit we extract will require many samples (as many pairs would be discarded); and (b) even if $\mathcal{D}$ has large entropy, the MA distribution tester may toss a large number of coins, and so we shall need to draw many samples accordingly.

The first concern can be easily handled by observing that distributions with small entropy can be efficiently *learned*, and so we can test them with few samples, even without the aid of a prover. Dealing with the second concern is significantly more involved, and requires proving a randomness reduction lemma for MA distribution testers, which shows that it suffices to extract a *small* number of uniformly random bits, roughly logarithmic in the domain size.

The proof of the aforementioned randomness reduction lemma follows the randomness reduction approach of Goldreich and Sheffet [35], but our different setting requires several new ideas. In particular, our model involves testers that access a proof and two sources of randomness and, most significantly, the argument in [35] crucially relies on a bound on the number of inputs that the tester can receive, but no such bound exists in our setting.

Acknowledgements. We are grateful to Oded Goldreich and Rocco Servedio for multiple technical and conceptual suggestions that greatly improved the results of this work and extended its scope. We thank Clément Canonne for many discussions concerning distribution testing and for offering advice regarding several specific topics. We thank Igor Shinkar and Nicholas Spooner for useful discussions.

References

1 Scott Aaronson and Avi Wigderson. Algebrization: A new barrier in complexity theory. *ACM Transactions on Computation Theory*, 1:2:1–2:54, 2009.

2 Jayadev Acharya, Clément L. Canonne, and Gautam Kamath. A chasm between identity and equivalence testing with conditional queries. In *Proceedings of the 19th International Workshop on Randomization and Computation*, RANDOM '15, pages 449–466, 2015.

3 Jayadev Acharya, Hirakendu Das, Ashkan Jafarpour, Alon Orlitsky, and Shengjun Pan. Competitive closeness testing. In *Proceedings of the 24th Annual Conference on Learning Theory*, COLT 2011, pages 47–68, 2011.

4 László Babai, Peter Frankl, and Janos Simon. Complexity classes in communication complexity theory. In *Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science*, FOCS 1986, pages 337–347, 1986.

5 László Babai and Shlomo Moran. Arthur-merlin games: a randomized proof system, and a hierarchy of complexity classes. *Journal of Computer and System Sciences*, 36:254–276, 1988.

6 Tuğkan Batu, Sanjoy Dasgupta, Ravi Kumar, and Ronitt Rubinfeld. The complexity of approximating the entropy. *SIAM Journal on Computing*, 35(1):132–150, 2005.

7 Tuğkan Batu, Eldar Fischer, Lance Fortnow, Ravi Kumar, Ronitt Rubinfeld, and Patrick White. Testing random variables for independence and identity. In *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science*, FOCS 2001, pages 442–451, 2001.

8 Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. Testing that distributions are close. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, FOCS 2000, pages 259–269, 2000.

9 Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. Robust PCPs of proximity, shorter PCPs, and applications to coding. *SIAM Journal on Computing*, 36(4):889–974, 2006.

10 Itay Berman, Ron D. Rothblum, and Vinod Vaikuntanathan. Zero-knowledge proofs of proximity. In *Proceedings of the 9th Innovations in Theoretical Computer Science Conference*, ITCS 2018, page To appear, 2018.

11 Bhaswar B. Bhattacharya and Gregory Valiant. Testing closeness with unequal sized samples. In *Proceedings of the 2015 Conference on Neural Information Processing Systems*, NIPS 2015, pages 2611–2619, 2015.

12 Arnab Bhattacharyya, Eldar Fischer, Ronitt Rubinfeld, and Paul Valiant. Testing monotonicity of distributions over general partial orders. In *Proceedings of the 2nd Innovations in Theoretical Computer Science Conference*, ITCS 2011, pages 239–252, 2011.

13 Eric Blais, Clément L. Canonne, and Tom Gur. Distribution testing lower bounds via reductions from communication complexity (Alice and Bob don't talk to each other anymore.). In *Proceedings of the 32th Conference on Computational Complexity*, CCC 2017, pages 1–42, 2017.

14 Kenneth P Burnham and David R Anderson. *Model selection and multimodel inference: a practical information-theoretic approach.* Springer Science & Business Media, 2003.

15 Clément Canonne and Ronitt Rubinfeld. Testing probability distributions underlying aggregated data. In *International Colloquium on Automata, Languages, and Programming*, ICALP '14, pages 283–295, 2014.

16 Clément L. Canonne. A survey on distribution testing. your data is big. But is it blue?, 2017.

17 Clément L. Canonne, Elena Grigorescu, Siyao Guo, Akash Kumar, and Karl Wimmer. Testing k-monotonicity. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:136, 2016. URL: `http://eccc.hpi-web.de/report/2016/136`.

18 Clément L. Canonne, Dana Ron, and Rocco A. Servedio. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 44(3):540–616, 2015.

19 Amit Chakrabarti, Graham Cormode, Andrew McGregor, and Justin Thaler. Annotations in data streams. *ACM Transactions on Algorithms*, 11, 2014.

20 Amit Chakrabarti, Graham Cormode, Andrew McGregor, Justin Thaler, and Suresh Venkatasubramanian. Verifiable stream computation and Arthur-Merlin communication. In *Proceedings of the 30th Conference on Computational Complexity*, CCC 2015, pages 217–243, 2015.

21 Sourav Chakraborty, Eldar Fischer, Yonatan Goldhirsh, and Arie Matsliah. On the power of conditional samples in distribution testing. *SIAM Journal on Computing*, 45(4):1261–1296, 2016.

22 Siu-On Chan, Ilias Diakonikolas, Gregory Valiant, and Paul Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the 25th Symposium on Discrete Algorithms*, SODA 2014, pages 1193–1203, 2014.

23 Alessandro Chiesa and Tom Gur. Proofs of proximity for distribution testing. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:155, 2017. URL: `https://eccc.weizmann.ac.il/report/2017/155`.

24 Ilias Diakonikolas and Daniel M. Kane. A new approach for testing properties of discrete distributions. In *Proceedings of the 57th Annual Symposium on Foundations of Computer Science*, FOCS 2016, pages 685–694, 2016.

25 Funda Ergün, Ravi Kumar, and Ronitt Rubinfeld. Fast approximate probabilistically checkable proofs. *Information and Computation*, 189(2):135–159, 2004.

26 Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Faster algorithms for testing under conditional sampling. In *Conference on Learning Theory*, COLT '15, pages 607–636, 2015.

27 Eldar Fischer, Yonatan Goldhirsh, and Oded Lachish. Partial tests, universal tests and decomposability. In *Proceedings of the 5th Innovations in Theoretical Computer Science Conference*, ITCS 2014, pages 483–500, 2014.

28 Eldar Fischer, Oded Lachish, and Yadu Vasudev. Trading query complexity for sample-based testing and multi-testing scalability. In *Proceedings of the 56th Symposium on Foundations of Computer Science*, FOCS 2015, pages 1163–1182, 2015.

29 Oded Goldreich. Introduction to property testing, 2017.

30 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998.

31 Oded Goldreich and Tom Gur. Universal locally verifiable codes and 3-round interactive proofs of proximity for CSP. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:192, 2016. URL: `http://eccc.hpi-web.de/report/2016/192`.

32 Oded Goldreich, Tom Gur, and Ilan Komargodski. Strong locally testable codes with relaxed local decoders. In *Proceedings of the 30th Conference on Computational Complexity*, CCC 2015, pages 1–41, 2015.

33 Oded Goldreich, Tom Gur, and Ron D. Rothblum. Proofs of proximity for context-free languages and read-once branching programs. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming*, ICALP 2015, pages 666–677, 2015.

34 Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography*, pages 68–75. Springer, 2011.

35 Oded Goldreich and Or Sheffet. On the randomness complexity of property testing. *Computational Complexity*, 19(1):99–133, 2010.

36 Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on computing*, 18(1):186–208, 1989.

37 Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the 18th a Annual ACM Symposium on Theory of Computing*, STOC 1986, pages 59–68, 1986.

38 Tom Gur and Ran Raz. Arthur-Merlin streaming complexity. *Information and Computing*, 243:145–165, 2015.

39 Tom Gur and Ron Rothblum. Non-interactive proofs of proximity. *Computational Complexity*, To appear, 2017.

40 Tom Gur and Ron D. Rothblum. A hierarchy theorem for interactive proofs of proximity. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, ITCS 2017, 2017.

41 Yael Tauman Kalai and Ron D. Rothblum. Arguments of proximity. In *Proceedings of the 35th Annual International Cryptology Conference*, CRYPTO 2015, pages 422–442, 2015.

42 Hartmut Klauck. On Arthur Merlin games in communication complexity. In *Proceedings of the 26th Conference on Computational Complexity*, CCC 2017, pages 189–199, 2011.

43 Erich L Lehmann and Joseph P Romano. *Testing statistical hypotheses*. Springer Science & Business Media, 2006.

44 Reut Levi, Dana Ron, and Ronitt Rubinfeld. Testing properties of collections of distributions. *Theory of Computing*, 9:295–347, 2013.

45 Ilan Newman and Mario Szegedy. Public vs. private coin flips in one round communication games. In *Proceedings of the 28th a Annual ACM Symposium on Theory of Computing*, STOC 1996, pages 561–570, 1996.

46 Liam Paninski. Estimating entropy on m bins given fewer than m samples. *IEEE Transactions on Information Theory*, 50(9):2200–2203, 2004.

47 Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10):4750–4755, 2008.

48 Aditi Raghunathan, Gregory Valiant, and James Zou. Estimating the unseen from multiple populations. *CoRR*, abs/1707.03854, 2017. `arXiv:1707.03854`.

49 Sofya Raskhodnikova, Dana Ron, Amir Shpilka, and Adam Smith. Strong lower bounds for approximating distribution support size and the distinct elements problem. *SIAM Journal on Computing*, 39:813–842, 2009.

50 Ran Raz and Amir Shpilka. On the power of quantum proofs. In *Proceedings of the 19th Conference on Computational Complexity*, CCC 2004, pages 260–274, 2004.

51 Omer Reingold, Ron Rothblum, and Guy Rothblum. Constant-round interactive proofs for delegating computation. In *Proceedings of the 48th ACM Symposium on the Theory of Computing*, STOC 2016, pages 49–62, 2016.

52 Guy N. Rothblum, Salil P. Vadhan, and Avi Wigderson. Interactive proofs of proximity: delegating computation in sublinear time. In *Proceedings of the 45th Symposium on Theory of Computing*, STOC 2013, pages 793–802, 2013.

53 Ronitt Rubinfeld. Taming big probability distributions. *ACM Crossroads*, 19(1):24–28, 2012.

54 Ronitt Rubinfeld and Rocco Servedio. Testing monotone high-dimensional distributions. *Random Structures & Algorithms*, 34:24–44, 2009.

55 Ronitt Rubinfeld and Madhu Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.

56 Gregory Valiant and Paul Valiant. Estimating the unseen: an $n/\log(n)$-sample estimator for entropy and support size, shown optimal via new CLTs. In *Proceedings of the 43rd Symposium on Theory of Computing*, STOC 2011, pages 685–694, 2011.

57 Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. *SIAM Journal on Computing*, 46(1):429–455, 2017.

58 Paul Valiant. Testing symmetric properties of distributions. *SIAM Journal on Computing*, 40:1927–1968, 2011.

59 Thomas Vidick and John Watrous. Quantum proofs. *Foundations and Trends in Theoretical Computer Science*, 11:1–215, 2016.

60 John Von Neumann. Various techniques used in connection with random digits. *National Bureau of Standards Applied Math Series*, 12:36–38, 1951.

61 John Watrous. Succinct quantum proofs for properties of finite groups. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, FOCS 2000, pages 537–546, 2000.

62 James Zou, Gregory Valiant, Paul Valiant, Konrad Karczewski, Siu On Chan, Kaitlin Samocha, Monkol Lek, Shamil Sunyaev, Mark Daly, and Daniel G MacArthur. Quantifying unobserved protein-coding variants in human populations provides a roadmap for large-scale sequencing projects. *Nature Communications*, 7, 2016.