Distribution-Free Proofs of Proximity

Hugo Aaronson
Department of Computer Science and Technology, University of Cambridge, UK

Tom Gur
Department of Computer Science and Technology, University of Cambridge, UK

Ninad Rajgopal
Department of Computer Science and Technology, University of Cambridge, UK

Ron D. Rothblum
Faculty of Computer Science, Technion, Haifa, Israel

Abstract

Motivated by the fact that input distributions are often unknown in advance, distribution-free property testing considers a setting in which the algorithmic task is to accept functions $f : [n] \to \{0, 1\}$ having a certain property Π and reject functions that are ε-far from Π, where the distance is measured according to an arbitrary and unknown input distribution $\mathcal{D} \sim [n]$. As usual in property testing, the tester is required to do so while making only a sublinear number of input queries, but as the distribution is unknown, we also allow a sublinear number of samples from the distribution $\mathcal{D}$.

In this work we initiate the study of *distribution-free interactive proofs of proximity* (df-IPPs) in which the distribution-free testing algorithm is assisted by an all powerful but untrusted prover. Our main result is that for any problem $\Pi \in \mathsf{NC}$, any proximity parameter $\varepsilon > 0$, and any (trade-off) parameter $\tau \leq \sqrt{n}$, we construct a df-IPP for Π with respect to ε, that has query and sample complexities $\tau + O(1/\varepsilon)$, and communication complexity $\tilde{O}(n/\tau + 1/\varepsilon)$. For τ as above and sufficiently large ε (namely, when $\varepsilon > \tau/n$), this result matches the parameters of the best-known general purpose IPPs in the standard uniform setting. Moreover, for such τ, its parameters are optimal up to poly-logarithmic factors under reasonable cryptographic assumptions for the same regime of ε as the uniform setting, i.e., when $\varepsilon \geq 1/\tau$.

For smaller values of ε (i.e., when $\varepsilon < \tau/n$), our protocol has communication complexity $\Omega(1/\varepsilon)$, which is worse than the $\tilde{O}(n/\tau)$ communication complexity of the uniform IPPs (with the same query complexity). With the aim of improving on this gap, we further show that for IPPs over specialised, but large distribution families, such as sufficiently smooth distributions and product distributions, the communication complexity can be reduced to $\tilde{O}(n/\tau^{1-o(1)})$. In addition, we show that for certain natural families of languages, such as symmetric and (relaxed) self-correctable languages, it is possible to further improve the efficiency of distribution-free IPPs.

2012 ACM Subject Classification Theory of computation → Interactive proof systems

Keywords and phrases Property Testing, Interactive Proofs, Distribution-Free Property Testing

Digital Object Identifier 10.4230/LIPIcs.CCC.2024.24

Related Version *Full Version*: `https://eccc.weizmann.ac.il/report/2023/118` [1]

Funding Tom Gur and Ninad Rajgopal are supported by the Tom Gur's UKRI Future Leaders Fellowship MR/S031545/1. Tom Gur is also supported in part by EPSRC New Horizons Grant EP/X018180/1 and EPSRC RoaRQ Grant EP/W032635/1. Ron Rothblum is funded by the European Union (ERC, FASTPROOF, 101041208). Part of this work was completed when the first three authors were affiliated with the University of Warwick. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Acknowledgements We are grateful to Oded Goldreich for his insightful comments, some of which led to rephrasing Theorem 1 to indicate the trade-off between the query and communication complexities with better clarity. We are also thankful to Marcel Dall'Agnol for many helpful discussions.

39th Computational Complexity Conference (CCC 2024).
Editor: Rahul Santhanam; Article No. 24; pp. 24:1–24:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Property Testing, initiated in [45, 25], is a rich and well-studied research field lying at the heart of many advancements in sublinear algorithms and complexity theory; see [21, 7] for a detailed introduction. Loosely speaking, a testing algorithm for a property Π is given oracle access to an input function $f : [n] \to \{0, 1\}$ and should decide whether $f \in \Pi$ using a small *sublinear* number of queries. As we cannot expect to do so exactly, the tester is required to distinguish between inputs that are in Π from those that are ε-far from every function in Π. Here, distance is typically measured using the relative Hamming distance – namely, the fraction of outputs of f that need to be changed to reach a member of Π.

While modeling distance using the relative Hamming distance is natural and convenient, in many settings it may not capture the underlying question (for example, when functions always satisfy a particular format or when some parts in the domain are more important than others). Following the Probably-Approximately-Correct (PAC) learning model, introduced by Valiant in his celebrated work in computational learning theory [49], *distribution-free* algorithms have widely been accepted as a closer abstraction of real-world computational tasks that are required to make decisions based on limited access to the input data. In this spirit, [25] introduced *distribution-free property testing*, where the distance between two functions is with respect to a distribution $\mathcal{D}$ (over inputs to the function), which is *arbitrary* and *unknown* to the testing algorithm. Since $\mathcal{D}$ is unknown, in addition to the query oracle to the input $f : [n] \to \{0, 1\}$, the tester can draw independent identically distributed random labelled samples $(i, f(i))$ from a *sample oracle*, where each index i is generated independently from the distribution $\mathcal{D}$. The tester is required to reject any function that is ε-far[1] from Π along the unknown distribution $\mathcal{D}$, and the only access that the tester has to $\mathcal{D}$ is via the sample oracle.

The distribution-free model of testing naturally complements the PAC-learning model, and profound bidirectional connections are known between them.[2] Moreover, distribution-free testing is motivated by the fact that it captures the realistic setting where the tester is required to maintain its guarantees despite dealing with data from an unknown environment (i.e., via data samples from some unknown and arbitrary distribution $\mathcal{D}$). It also deals with situations where not all underlying data points are equally important, e.g., in graphs where certain edges or vertices are more important than others, and one would like to consider distributions that weigh them appropriately.

Following [25], several distribution-free testing algorithms have been designed for function classes including monotone Boolean functions and low-degree polynomials over finite fields [33], k-juntas [37, 11, 3], conjunctions (monotone or non-monotone) and linear threshold functions [19, 13], polynomial threshold functions and decision trees [8], halfspaces [8, 12], and low-degree polynomials on $\mathcal{R}^n$ [18, 2]. Distribution-free testing has also been studied for graph properties including connectivity [34], bipartiteness [22], k-path and degree regularity [23], as well as for word problems like subsequence-freeness [41].

Despite such strides of progress, our understanding of distribution-free testing is much more limited than that of testing with respect to the uniform distribution. This is due to the multitude of challenges that arise in designing algorithms that need to deal with data samples that can come from any arbitrary distribution, which in turn, makes the model significantly more involved.

[1] We say $f : [n] \to \{0, 1\}$ is ε-far from a (non-empty) property Π along $\mathcal{D}$, if for every $f' : [n] \to \{0, 1\}$ such that $f' \in \Pi$, it holds that $\mathbb{P}_{i\sim\mathcal{D}}[f(i) \neq f'(i)] > \varepsilon$.

[2] In particular, in [25], it is shown that if a class of functions $\mathcal{C}$ has a *proper* PAC-learner using membership queries (where the learner outputs an approximate hypothesis that also belongs to $\mathcal{C}$), then $\mathcal{C}$ has a distribution-free tester that uses roughly the same number of queries and samples as the learner.

This paper aims to bridge the gap between testing over the uniform distribution and distribution-free testing by capitalising on the power of interactive proofs, and delegating the task of handling the challenges imposed by the distribution-free setting to a powerful, but untrusted, prover.

1.1 Distribution-free Interactive Proofs of Proximity

In this work, we initiate the study of *distribution-free interactive proofs of proximity* (distribution-free IPPs), which are distribution-free testers that are augmented with the help of a prover. In the rest of this paper, for convenience, rather than thinking of the input as a function, we view it as a string $x \in \{0,1\}^n$ (which can be similarly be viewed as a truth table of a function $f_x : [n] \to \{0,1\}$). Correspondingly, we view a property Π of functions as a language L over strings (which may be viewed as truth tables of the functions in Π).

Thus, distribution-free IPPs are protocols where a *sublinear* time, randomised algorithm, called the verifier, interacts with an untrusted prover to decide whether the given input $x \in \{0,1\}^n$ belongs to the language L or is far from such, where distance is measured with respect to a fixed, but unknown distribution $\mathcal{D}$ over $[n]$. The verifier is given access to the input x through a query oracle, as well as a sample oracle with respect to $\mathcal{D}$, while the prover can look at the input entirely. We assume that the prover does not know the queries that the verifier makes to either of its oracles.

We require that for any $x \in L$, there exists an honest prover that interacts with the verifier and convinces it to accept with high probability, while when x is ε-far from L with respect to the distribution $\mathcal{D}$, no cheating prover, even computationally unbounded, will make the verifier accept, except with low probability. Further, we require the distribution-free IPP to meet these requirements, with respect to the underlying (and unknown) distribution $\mathcal{D}$ from which the oracle draws samples.

In this setting, the verifier's *query complexity* and *sample complexity*, the number of bits exchanged in the protocol, i.e., the *communication complexity*, and the verifier's running time should all be sublinear in input length. Other complexity parameters of interest are the number of rounds of interaction, and the (honest) prover's running time.

Distribution-free IPPs capture the distribution-free property testing analogue of interactive proofs (for more information, see Section 1.4). As such, similar to uniform IPPs, distribution-free IPPs can be alternatively viewed as proof systems where the bounded verifier need only be convinced of the fact that the input is close to the language, by interacting with a more powerful prover. One of the main goals of distribution-free IPPs is to overcome the inherent limitations of distribution-free testing algorithms by showing that for certain properties, verifying proximity over arbitrary distributions is considerably faster with a prover than actually testing it. In particular, we want to design distribution-free IPPs (with sublinear query complexity) for rich families of properties that have no known distribution-free testers.

Of close relevance are the well-studied notion of IPPs over the uniform distribution, which we refer to in this work as Uniform IPPs, that were introduced in [16, 44] (and are trivially generalised by distribution-free IPPs). Showcasing the power of interaction, [44] constructed highly non-trivial uniform IPPs for every language that can be decided in bounded depth (e.g., NC), which was recently made near-optimal by [43] (see [36] for the conditional matching lower bound), and strengthened to encompass also bounded space languages [40].

Motivated intrinsically and by natural applications to *delegation of computation*, the study of uniform IPPs has drawn much recent attention on its own right [44, 32, 36, 40, 26, 20]. Moreover, their study has led to interesting models and applications of sublinear time verification, including non-interactive proofs of proximity (or MAPs) [32] (a related model was studied concurrently and independently by [17]), arguments of proximity [36], testing properties of distributions [14, 35], interactive oracle proofs of proximity [40, 4, 42, 10], verifying machine learning tasks [29], batch verification for UP [39, 43], as well as variants involving zero-knowledge [6] and quantum computation [15].

1.2 Our Results

Our main contribution is constructing distribution-free IPPs for any language in NC, which for any query vs communication trade-off parameter $\tau \leq \sqrt{n}$, matches the complexity of the best known IPPs for most settings of the proximity parameter ε – specifically, when $\varepsilon \geq \tau/n$. We further improve the efficiency of distribution-free IPPs for general ε (i.e., when $\varepsilon < \tau/n$), under specific distribution families such as "smooth" and "learnable" distributions, which are defined below.

In addition, for certain families of languages, such as symmetric and relaxed self-correctable languages, we construct distribution-free IPPs that improve on our general-purpose distribution-free IPPs, then use them to provide separation results that provide further insight into the distribution-free IPP model.

We elaborate on these results next.

1.2.1 Distribution-free IPPs for NC

Our first main result is a sublinear distribution-free IPP for any language computable by low-depth circuits. In more detail, let (logspace-uniform) NC be the set of languages computable by (logspace-uniform) Boolean circuits of polynomial size and poly-logarithmic depth. We show that every language in NC has a distribution-free IPP with sublinear complexity measures, for almost all values of the proximity parameter ε. We emphasize that this is in stark contrast to distribution-free testers, which are only known for a handful of languages based on their combinatorial or algebraic structure. Indeed, the following theorem shows that distribution-free IPPs capture a much richer class of languages that need not have such special structural properties.

▶ **Theorem 1 (Distribution-Free IPP for NC).** *For every language L in logspace-uniform* NC *and every trade-off parameter $\tau = \tau(n) \leq \sqrt{n}$, there exists a distribution-free* IPP *for L with proximity parameter $\varepsilon \geq \Omega\left(\frac{\log^3(n)}{n}\right)$, query complexity $\tau + O\left(\frac{1}{\varepsilon}\right)$, sample complexity $\tau + O\left(\frac{1}{\varepsilon}\right)$ and communication complexity $\tilde{O}\left(\frac{n}{\tau} + \frac{1}{\varepsilon}\right)$.*

Moreover, the verifier runs in time $\tilde{O}\left(\frac{n}{\tau} + \frac{1}{\varepsilon}\right)$, the prover runs in time poly(n) *and the round complexity is* poly*log*(n).

Here, τ denotes the parameter that trades-off between the query and communication complexities of the distribution-free IPP. Note that, for the above values of τ, our distribution-free IPP has sublinear query and communication complexity even for very small values of the proximity parameter ε of the form $1/n^{1-\delta}$, where $\delta > 0$. An interesting instantiation of our result is obtained by setting τ to $\sqrt{n}$, and thus, for every $\varepsilon \geq 1/\sqrt{n}$, the query complexity and sample complexities are $O(\sqrt{n})$, while the communication complexity and verifier running times are both $\tilde{O}(\sqrt{n})$.

It is worth noting that, for every $\varepsilon \geq \frac{1}{\tau}$ (and $\tau \leq \sqrt{n}$), this result is conditionally optimal up to poly-logarithmic factors, since [36] show a lower bound of $\Omega(n)$ on the product of the query and communication complexities of a uniform IPP for a language in NC^1, under a strong, but reasonable, cryptographic assumption. Furthermore, for any ε, the query complexity of $\Omega(1/\varepsilon)$ is necessary for any IPP over non-degenerate languages, even over the uniform distribution (see [44, Remark 1.2]).

▶ Remark 2. While Theorem 1 refers to distribution-free IPPs over NC languages, the theorem can be made more general. In particular, it also yields distribution-free IPPs with sublinear query and communication complexities for languages computable by circuits of sub-exponential size and bounded polynomial depth.

Likewise, in a similar fashion to the known literature on uniform IPPs, we can combine our techniques directly with [40] to get a constant-round distribution-free IPP for any language that is computable in $\mathsf{poly}(n)$ time and bounded polynomial space.

Comparison to Uniform IPPs for NC [44, 43]

For any language in NC, Rothblum, Vadhan and Wigderson [44] construct a uniform IPP for any $\tau = \tau(n)$ and proximity parameter $\varepsilon > 0$, with query complexity $\tau + O(1/\varepsilon)^{1+o(1)}$ and communication complexity $\frac{n}{\tau^{1-o(1)}}$. Rothblum and Rothblum [43] improve on this, by reducing the communication complexity to $\frac{n}{\tau} \cdot \mathsf{poly}log(n)$. In particular, the latter obtains an optimal trade-off, up to poly-logarithmic factors, between the query and communication complexities of a uniform IPP (conditionally, from [36]), for every value of τ and $\varepsilon \geq 1/\tau$. While these results are stated in [44, 43] by implicitly setting $\tau = O(1/\varepsilon)$, for any given ε, this IPP formulation parameterised by τ is obtained by inspection (see also [26, Theorem 6.3]). For comparison, in this setting, our distribution-free IPP has the same query (and sample) complexity, while the communication complexity and verifier running times are both $\tilde{O}(\varepsilon \cdot n + 1/\varepsilon)$.[3]

Theorem 1 gives a construction of a *distribution-free* IPP for any NC language that matches the query and communication complexities of the uniform IPP by [43], when $\varepsilon \geq \tau/n$. Moreover, this obtains the (conditionally) optimal trade-offs between query and communication complexities in the *same regime* of ε, but when $\tau \leq \sqrt{n}$. Indeed, when $\varepsilon \geq 1/\tau$, the product of the query and communication complexities of the distribution-free IPP from Theorem 1 is $\tilde{O}(n + \tau^2)$. Our protocol builds on [44], introducing new ideas that allow us to construct IPPs in the more involved distribution-free setting.

Finally, when the proximity parameter ε is very small, Theorem 1 suffers a blow-up in the communication complexity compared to the uniform IPPs of [44, 43]. In more detail, when $\varepsilon \ll \tau/n$, the communication complexity in our distribution-free IPP is $\tilde{\Omega}\left(\frac{1}{\varepsilon}\right)$, whereas the communication complexity achieved by the uniform IPPs is $\tilde{O}\left(\frac{n}{\tau}\right)$ (the query complexity roughly remains the same across all three cases). Thus, our distribution-free IPP has communication complexity at least $\Omega(n/\tau)$ for every value of ε, whereas the communication complexity of the uniform IPPs is much lower when $\varepsilon \ll \tau/n$.

1.2.2 IPPs for NC: The case of small ε

Following the discussion in the last section, we aim to construct distribution-free IPPs that achieve query and communication complexities that match the state-of-the-art uniform IPP for every value of ε. While we unable to do so in the most general case, we construct such IPPs over *specific families of distributions*, which match the complexities of [44] and, in turn, differ from the complexities of [43] only by a factor of $n^{o(1)}$. For these IPPs, while the underlying distribution is still unknown, it is guaranteed to come from the specific family of distributions under consideration.

[3] In fact, we prove that for every value of the parameter τ and ε, the distribution-free IPP from Theorem 1 has communication complexity $\tilde{O}(\tau + n/\tau + 1/\varepsilon)$; thus, setting $\tau = O(1/\varepsilon)$ suffices. An additional point to note is that when $\tau > \sqrt{n}$, the IPP always has worse communication complexity than its uniform counterpart irrespective of the value of ε, and further, never meets the optimal [36] lower bound. As such, we only consider $\tau \leq \sqrt{n}$ as a more interesting regime of study.

To describe our results, it will be convenient throughout this section to identify $[n]$ with the elements of an m-dimensional tensor of size $k \in \mathcal{N}$ in each dimension, such that $k^m = n$. In such a case, we refer to $[n]$ as $[k]^m$ (by fixing some canonical bijection between them).

ρ-Dispersed Distributions

Intuitively speaking, ρ-dispersed distributions capture the sense that for a smooth distribution over $[k]^m$, along any dimension, its probability mass on any element in $[k]^m$ is not much larger than the average of the probability masses of its neighbours. ρ-dispersed distributions relax this requirement by having the probability mass on any element bounded by ρ times the expected mass on any of its neighbours.[4]

We show that for distributions that are reasonably smooth in this sense, i.e. for ρ-dispersed distributions for $\rho \leq k^{o(1)}$, we obtain IPPs for NC over such distributions for every $\tau = \tau(n) < n$ and $\varepsilon > 0$, with query complexity $O(\tau + 1/\varepsilon)^{1+o(1)}$, and communication complexity of $\tilde{O}\left(\frac{n}{\tau} \cdot \tau^{o(1)}\right)$, thus matching the bounds obtained by [44]. It is worth noting that $k^{o(1)}$-dispersed distributions are still quite general, e.g. any distribution where the probability mass on any element in $[k]^m$ is in the range $\left[\frac{1}{an}, \frac{a}{n}\right]$, for some $a \leq k^{o(1)}$ is $k^{o(1)}$-dispersed.

▶ **Theorem 3** (**IPP for NC over ρ-dispersed distributions**). *For every language in logspace-uniform* NC*, every $m, n, k \in \mathcal{N}$ such that $m = \log_k(n)$ (i.e., $k^m = n$) and $\rho \in \mathcal{R}$ such that $\rho \leq k^{o(1)}$, for every proximity parameter $\varepsilon > 0$ and trade-off parameter $\tau > 0$, there exists an* IPP *over ρ-Dispersed distributions over $[k]^m$ with query and sample complexities $O(\tau + 1/\varepsilon)^{1+o(1)}$ and communication complexity $\tilde{O}\left(\frac{n}{\tau^{1-o(1)}}\right)$.*

Moreover, the verifier runs in time $n^{o(1)} \cdot \left(\tau + \frac{n}{\tau} + \frac{1}{\varepsilon}\right)$, the prover runs in time poly(n) *and the round complexity is* poly*log*(n).

Theorem 3 also holds generally over ρ-dispersed distributions, for any ρ. The query complexity increases with ρ, while the communication complexity is *independent of ρ*. Theorem 3 builds on the ideas used for the distribution-free IPP from Theorem 1 while incorporating new technical insights into the analysis by [44] to generalise over ρ-dispersed distributions. We leave the task of obtaining IPPs over ρ-dispersed distributions that match [43] as future work.

1.2.2.1 Product Distributions in the White-Box model

Note that in the IPPs of Theorems 1 and 3, the verifier does not learn the underlying distribution $\mathcal{D}$. Hence, we ask the following question: if we could gain more information about $\mathcal{D}$, or further, learn a reasonably good approximation for $\mathcal{D}$, can we improve the query complexity of the IPPs, over general values of ε? We answer this question in the affirmative for product distributions.

We consider the *white-box model* for distribution-free IPPs, where the verifier receives a succinct description of the unknown distribution $\mathcal{D}$ over $[k]^m$ via a *polynomial-sized* sampling circuit C, in addition to query access to the input string. It is worth noting that, for white-box IPPs, the sample complexity is irrelevant since the verifier has a succinct description of the entire distribution. Thus, the main complexity parameters here are the query complexity, communication complexity, and the verifier running time.

[4] For example, the uniform distribution is the only 1-dispersed distribution, i.e., a maximally smooth distribution in this sense. On the other hand, every distribution over $[k]^m$ is trivially a k-dispersed distribution.

While white-box models have been widely studied in the setting of zero-knowledge proofs [46, 48, 47] and in distribution testing (see survey by [27]), we use this model to construct IPPs for languages in NC over a generalised family of product distributions over $[k]^m$, to get improved complexities for general values of ε, compared to the distribution-free IPP from Theorem 1. We call this family as *m-product distributions*, and denote any such distribution $\mathcal{D}$ as $\mathcal{D} = \mathcal{D}_1 \times \dots \mathcal{D}_m$, where each $\mathcal{D}_j$ is supported on $[k]$ and is independent of any other coordinate distributions. In particular, $\mathcal{D}(i_1, \dots, i_m)$ is defined as $\prod_{j=1}^{m} \mathcal{D}_j(i_j)$.

▶ **Theorem 4 (IPPs for NC over m-product distributions).** *For every language in logspace-uniform* NC*, every $\tau = \tau(n)$, $\varepsilon > 0$, and $m, n, k \in \mathcal{N}$ such that $m \leq \log(n)$ and $k^m = n$, there exists a white-box* IPP *for L over m-product distributions over $[k]^m$. The* IPP *has query complexity $O(\tau + 1/\varepsilon)^{1+o(1)}$ and communication complexity $\left(\frac{n}{\tau^{1-o(1)}} \cdot k + k^2\right) \cdot \mathsf{poly}log(n)$. Moreover, the verifier runs in time $n^{o(1)}\left(\frac{n}{\tau} \cdot k + \tau + k^2 + \frac{1}{\varepsilon}\right)$ and the round complexity is* $\mathsf{poly}log(n)$.

When m is large enough (like $m = \log(n)$), the query and communication complexity trade-off, as well as the verifier running time of the IPP from Theorem 4 match that of the uniform IPP from [44], while working in this setting.[5] Theorem 4 builds on the framework of Theorem 1, and uses several new ideas in the construction of the IPP, as well as its analysis, to improve the complexity. Crucially, it uses that any product distribution has a succinct description to be able to *learn* it in the white-box-setting.

It is worth stressing that the IPPs from Theorems 3 and 4 are incomparable. Indeed, there exist m-product distributions $\mathcal{D} = \mathcal{D}_1 \times \cdots \times \mathcal{D}_m$ that are poorly dispersed, for eg., $\mathcal{D}$ is no longer smooth when some $\mathcal{D}_j$ has a large probability mass over just one element (one row or more generally, a few rows). For such distributions, the IPP from Theorem 4 provides a much better query and communication trade-off than the IPP from Theorem 3, which is a more general result for smooth distributions.

1.2.3 On the power of distribution-free IPPs

Recall that Theorems 3 and 4 improve the query and communication complexity trade-off of our general distribution-free IPP in Theorem 1, by considering special families of distributions to design the IPPs over. A natural direction that complements this approach is to ask whether we can use additional information about the *language* L instead, to construct super-efficient distribution-free IPPs.

In turn, we study distribution-free IPPs for specific problems of interest. On one hand, for certain problems we can hope to improve the various associated complexity parameters over our general distribution-free IPP by capitalising on the structure of the language. On the other hand, this allows us to obtain complexity-theoretic separations between the power of standard, non-interactive, and interactive distribution-free testers.

1.2.3.1 Symmetric languages

We study the power of distribution-free testers and IPPs for symmetric languages, which are languages that are invariant under permutations. We show that there exist symmetric languages that are hard for distribution-free testers, yet, given interaction with a prover, the symmetrical structure can be leveraged to obtain exponentially faster distribution-free IPPs.

[5] A subtle point here is that while Theorem 4 is over product distributions over $[k]^m$, when $m = 2$ (or a small constant), we get sublinear complexities only by considering distributions over biased matrices $[k_1] \times [k_2]$.

▶ **Theorem 5** (**Distribution-free IPPs for symmetric languages**). *The following statements hold.*

1. *Let L be a symmetric language. Then, there exist a distribution-free* IPP *for L with sample complexity $O(1/\varepsilon)$, communication complexity $O(\log^2(n)/\varepsilon)$ and $O(\log(n)/\varepsilon)$ round complexity.*
2. *There exists a symmetric language L' for every $\varepsilon > 0$ such that any distribution-free property tester for L' requires $\Omega(n^{1/3-0.0005})$ queries and labeled samples from the input.*

1.2.3.2 (Relaxed) self-correctable languages

Next, we show that for languages that admit self-correctability, we can transform any IPP into a distribution-free IPP at a negligible cost. In fact, we can deal with a far more general class of languages; namely, languages that are *relaxed locally correctable* [5, 31]. Loosely speaking, these are languages that admit a correcting algorithm that is required to correct the symbol at every location of the codeword, by reading a small number of locations in it, but is allowed to abort if noticing that the given word is corrupted. This family of languages is of central importance in the interactive proofs and probabilistically checkable proofs literature, and in particular, it captures languages of low-degree polynomials, holographic IPPs, and various relaxed locally correctable and decodable languages that were used to prove complexity-theoretic separations (cf. [30]).

▶ **Proposition 6** (**Generic Transformations for IPPs for RLCCs**). *For any subset L of a binary RLCC, $C \subseteq \{0,1\}^n$, if L has an* IPP *over the uniform distribution with query complexity q and communication complexity c for proximity $\varepsilon > 0$, then there exists a distribution-free* IPP *for L with the same round complexity, communication complexity and query complexity $q + O(\frac{t}{\varepsilon})$, where t is the query complexity of the corrector of C.*

As a corollary of Proposition 6, we are able to lift complexity-theoretic results concerning uniform IPPs to the setting of distribution-free IPPs. In particular, we obtain strong separations between the power of distribution-free testers, distribution-free non-interactive proofs of proximity (MAPs), and distribution-free IPPs.

▶ **Corollary 7** (**Complexity separations**). *There exists a language L such the following hold true.*

1. Property Testing*: The query complexity of distribution-free testing L (without a proof) is $\Theta(n^{0.999\pm o(1)})$.*
2. MAP*: L has a distribution-free* MAP *with query and communication complexities $\Theta(n^{0.499\pm o(1)})$. Moreover, for every $p \geq 1$, the distribution-free* MAP *query complexity of L with respect to proofs of length p is $\Theta\big(\frac{n^{0.999\pm o(1)}}{p}\big)$.*
3. IPP*: L has a distribution-free* IPP *with query and communication complexities* poly*log*(n).

Complementing this Corollary, we prove the existence of languages that can be tested under the uniform distribution with low query complexity (and thus, have a uniform IPP with low query complexity and no communication), but for which distribution-free IPPs require large query complexity or large communication complexity. This illustrates the difficulty of constructing distribution-free IPPs vs. standard uniform IPPs.

▶ **Proposition 8** (Distribution-free IPPs vs. uniform testing). *The following hold true:*

1. *There exists $\varepsilon > 0$ and a language L such that L has a property tester over the uniform distribution with query complexity $O(1/\varepsilon)$ for proximity parameter ε. However, for any distribution-free* MAP *for L with proximity parameter ε, query complexity q, and proof length p, $\max(q,p) = \Omega(\varepsilon \cdot n)$.*

2. *Assuming the existence of exponentially hard pseudo-random generators, there exists $\varepsilon > 0$ such that for all $q = q(n) \leq n$, there exists a language L, such that for any distribution-free* IPP *for L with proximity parameter ε, communication complexity c, and query complexity q, $\max(c, q) = \Omega(\sqrt{\varepsilon \cdot n})$. However, L has a uniform property tester with query complexity $O(1/\varepsilon)$ for proximity parameter ε.*

Table 1 provides a comparison of some of these results with related testing models. It is an interesting open direction to exhibit distribution-free IPPs that improve on the query complexity lower bounds known for distribution-testing functional properties like monotonicity [33], monotone conjunctions [13], or k-juntas [38].

Table 1 This is a table of our main results (TensorSum as defined in [32]). The complexities shown here are those that minimise the sum of the query and communication complexity. Note that while the uniform property tester for symmetric properties is more efficient than the corresponding uniform IPP, this only holds for restricted (constant) values of ε.

	Property Testing	IPP	DF-Property Testing	DF-IPP
Languages in NC	$\Omega(n)$ (e.g., low-degree univariate polynomial)	$\tilde{O}(\sqrt{n})$ [44, 43]	$\Omega(n)$ similarly	$\tilde{O}(\sqrt{n})$ (arbitrary distributions, for $\varepsilon \geq 1/\sqrt{n}$); see Theorem 1 $n^{1/2+o(1)}$ (smooth distributions); see Theorem 3 $n^{1/2+o(1)}$ (product distributions); see Theorem 4
TensorSum	$\Omega(n^{0.99+o(1)})$ [32]	poly$log(n)$ [32]	$\Omega(n^{0.99+o(1)})$ Trivially, from [32]	poly$log(n)$; see Corollary 7
Symmetric Properties	$\Theta(1)$ ($\varepsilon = O(1)$) Folklore	poly$log(n)$ [44]	$\Omega(n^{\frac{1}{3}})$ Theorem 5	poly$log(n)$; see Theorem 5

1.3 Technical Overview

In this technical overview, we highlight the proofs of Theorems 1, 3, and 4. The general strategy for proving these theorems builds on the Uniform IPPs for NC from [44, 43]. However, the setting of distribution-free testing is more involved, and below, we highlight the key challenges encountered in this setting, and our ideas to overcome them. Our distribution-free IPPs are constructed through an interplay of various techniques and tools from interactive proofs, property testing, and distribution testing.

Note that, for convenience, we show the construction of the distribution-free IPP from Theorem 1 in the setting of $\tau = O(1/\varepsilon)$, for any proximity parameter ε, obtaining query complexity $O(1/\varepsilon)$ and communication complexity $\tilde{O}(\varepsilon \cdot n + 1/\varepsilon)$. This can be shown to be equivalent to the statement of Theorem 1 that is parameterised by τ. Similarly, the IPPs for our other results are parameterised in terms of the proximity parameter ε. For detailed proofs, we refer the reader to the full version [1].

1.3.1 Proof outline of Theorem 1

The [44] protocol (as well as the follow-up work [43]) is centered around a parameterised problem called PVAL. Loosely speaking, the PVAL language contains all strings, whose encoding under a specific code, called the low degree extension, is equal to given values when projected on to the given coordinates. More precisely, the PVAL problem is parameterised by a (sufficiently large) finite field $\mathcal{F}$, integers k, m, n such that $k, m < |\mathcal{F}|$ and $k^m = n$, a set of vectors $J = (j_1, \ldots, j_t) \subset \mathcal{F}^m$ of size t and a t-length vector $\vec{v} \in \mathcal{F}^t$. An input $X \in \mathcal{F}^{k^m}$ is in $\mathsf{PVAL}(J, \vec{v})$ if it holds that $P_X(j_i) = v_i$, for every $i \in [t]$, where $P_X : \mathcal{F}^m \to \mathcal{F}$ is the m-variate low-degree extension (LDE) of X.[6]

The interactive reduction from NC to PVAL

Let L be any language in NC and let $\varepsilon > 0$ be the input proximity parameter. Let $X \in \{0,1\}^n$ be the input to L and $\mathcal{D}$ be the unknown underlying distribution over which the verifier can access X through a sample oracle. The first step in [44] is to show an interactive reduction Π_{NC} from L to (a parameterisation of) PVAL, where the verifier *does not access* the input $X \in \{0,1\}^n$.[7]

In more detail, let $B_{\mathcal{D}}(X)$ (respectively $B_{\mathcal{U}}(X)$) be the set of binary strings that are at a distance at most ε along the distribution $\mathcal{D}$ (respectively the uniform distribution $\mathcal{U}$) from X. In [44], the verifier in Π_{NC} generates parameters $(\mathcal{F}, k, m, J, \vec{v})$ for PVAL, where J is a set of t points in $\mathcal{F}^m$, such that the following hold when t is sufficiently large.

- If $X \in L$, then $X \in \mathsf{PVAL}(J, \vec{v})$.
- If X is ε-far from L along $\mathcal{U}$ then, with high probability over the verifier's randomness, $B_{\mathcal{U}}(X)$ and $\mathsf{PVAL}(J, \vec{v})$ are disjoint. In other words, with high probability, X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ along $\mathcal{U}$.

Furthermore, the points J output by the reduction Π_{NC} are *distributed uniformly at random* in $(\mathcal{F}^m)^t$. Crucially, [44] show that the guarantees over the outputs of this reduction *only hold* when $t = O(\log(|B_{\mathcal{U}}(X)|)$ many points are picked in J.[8]

Since the size of the set $B_{\mathcal{U}}(X)$ is $\binom{n}{\varepsilon n} \leq O(2^{\varepsilon n \log(n)})$, following from the earlier discussion, by setting $t = O(\log(|B_{\mathcal{U}}(X)|) = \tilde{O}(\varepsilon n)$, we ensure that the guarantees of Π_{NC} hold. An immediate attempt would be try to extend this analysis verbatim to distribution-free testing, by setting t to $O(\log(|B_{\mathcal{D}}(X)|))$ instead, and thus having Π_{NC} guarantee that X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ along the distribution $\mathcal{D}$, for soundness. However, for an arbitrary unknown distribution $\mathcal{D}$, the size of $B_{\mathcal{D}}(X)$ can be prohibitively large. For example, when $\mathcal{D}$ is supported over the first $\log(n)$ indices, for any value of ε, the size of $B_{\mathcal{D}}(X)$ blows up to at least $2^{n-\log(n)}$! Thus, for our choice of t, we already lose the sublinear time verification and communication complexity, and it is unclear if this reduction can achieve such soundness guarantees for PVAL.

Uniform IPP for PVAL is also "complete" for distribution-free IPPs for NC

Our key idea for constructing the distribution-free IPP for L, is in fact, an interactive reduction Π' to constructing a *uniform* IPP for PVAL (with a different parameterisation for PVAL than that obtained by Π_{NC}). Theorem 1 follows by using the ready-made uniform IPP for PVAL by [43].

[6] Recall that the m-variate LDE P_X is the unique polynomial with individual degree $k-1$ such that P_X agrees with X on $[k]^m$, where we identify $[k]$ with a subset of field elements in some canonical way.

[7] Technically, an interactive proof is specified by a verifier and an honest prover. However, for the sake of exposition we refer to them both together as Π_{NC} in this section.

[8] Π_{NC} runs t parallel copies of the interactive reduction from L to PVAL over a single point by [28], with the guarantee that if the input $X \notin L$, the probability that X is also in PVAL over t points, is at most 2^{-t}. Now, if X were instead ε-far from L, then a union bound over all the points in $B_{\mathcal{U}}(X)$ ensures a small probability for the event that there exists a point in $B_{\mathcal{U}}(X)$ that is also in PVAL over t points.

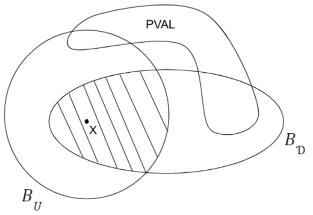

Figure 1 The shaded region $(B_{\mathcal{U}}(X) \cap B_{\mathcal{D}}(X))$ consists of the set of points in $\{0,1\}^n$ that are ε-close to X with respect to both $\mathcal{D}$ and $\mathcal{U}$. The soundness promise of the interactive reduction Π' ensures that any string in $\mathsf{PVAL}(J, \vec{v})$ is present in at most one of $B_{\mathcal{U}}(X)$ or $B_{\mathcal{D}}(X)$, but not in both (shaded region) (with high probability).

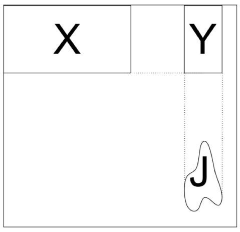

Figure 2 In the uniform IPP for PVAL, the prover sends the $(m-1)$-variate LDE of each row of X evaluated on J_2 (column indices of J), in the form of the purported matrix $Y' \in \mathcal{F}^{k \times t}$. However, to ensure consistency of Y' with respect to $\mathsf{PVAL}(J, \vec{v})$, for any $j = (a, b) \in J$, the univariate LDE of the b^{th}-column of Y' evaluated on a is required to be equal to $\vec{v}[j]$.

Consider a NO input $X \in \{0,1\}^n$ to L, that is, an input that satisfies the soundness requirement $d_{\mathcal{D}}(X, L) > \varepsilon$, over the unknown distribution $\mathcal{D}$. To start with, Π' runs the interactive reduction Π_{NC} from L to $\mathsf{PVAL}(J, \vec{v})$ with the same value of $t = |J| = \tilde{O}(\varepsilon n)$.

Setting t to be $O(\log(|B_{\mathcal{D}}(X) \cap B_{\mathcal{U}}(X)|)) \le O(\log(|B_{\mathcal{U}}(X)|)) = \tilde{O}(\varepsilon n)$, we can generalise the guarantees of Π_{NC} to show that the intersection of $\mathcal{B}_{\mathcal{U}}(X)$ and $\mathcal{B}_{\mathcal{D}}(X)$ is disjoint from $\mathsf{PVAL}(J, \vec{v})$, with high probability. Indeed, this builds on the earlier argument (and Footnote 8), but over $\mathcal{B}_{\mathcal{U}}(X) \cap \mathcal{B}_{\mathcal{D}}(X)$, alongside the fact that the size of this set is upper bounded by the size of $\mathcal{B}_{\mathcal{U}}(X)$. Thus, X cannot be ε-close to $\mathsf{PVAL}(J, \vec{v})$ along *both* $\mathcal{U}$ and $\mathcal{D}$, or in other words, X is ε-far from every element of PVAL along at least one of the two distributions (see Figure 1 for details).

Following this, assume that $d_{\mathcal{D}}(X, \mathsf{PVAL}(J, \vec{v})) > \varepsilon$. We construct the next stage of Π', based on a case analysis whether X is *additionally* ε-far from $\mathsf{PVAL}(J, \vec{v})$ under the uniform distribution or not. Indeed, suppose that X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ under the uniform distribution. This is the easy case; we can catch this with the uniform IPP for $\mathsf{PVAL}(J, \vec{v})$ as usual.

On the other hand, suppose that instead, X is close to $\mathsf{PVAL}(J, \vec{v})$ under the uniform distribution, i.e., $d_{\mathcal{U}}(X, \mathsf{PVAL}(J, \vec{v})) \le \varepsilon$. At this point, we observe (following [43]) that when J is distributed uniformly at random, with high probability $\mathsf{PVAL}(J, \vec{v})$ is a good error correcting code (i.e., with large minimal distance).[9] Since the output J of Π_{NC} is distributed uniformly at random, when X is ε-close to $\mathsf{PVAL}(J, \vec{v})$ over the uniform distribution, Π_{NC} guarantees that X is in fact close to a *unique* element X' in $\mathsf{PVAL}(J, \vec{v})$.

To summarize, so far we have that X is ε-close to $X' \in \mathsf{PVAL}(J, \vec{v})$ along $\mathcal{U}$, but by our soundness condition, X is ε-far from $\mathsf{PVAL}(J, \vec{v})$, and in particular from X', along $\mathcal{D}$. Now, the verifier uses the sample oracle to $\mathcal{D}$ to generate $O(1/\varepsilon)$ samples, which we denote by $I \subseteq [n]$, and the corresponding values in X given by $X|_I$. From the soundness assumption, with high probability there exists an index i in I such that $X_i \neq X'_i$. Combining this with the fact that every element in $\mathsf{PVAL}(J, \vec{v})$ other than X' is ε-far from X along the uniform distribution, X' is not in $\mathsf{PVAL}((J, I), (\vec{v}, X|_I))$, where PVAL is parameterised over a larger set. In other words, we see that X is ε-far from $\mathsf{PVAL}((J, I), (\vec{v}, X|_I))$ along the *uniform distribution* and a uniform IPP for $\mathsf{PVAL}((J, I), (\vec{v}, X|_I))$ suffices.

[9] It is worth emphasising that this does not hold for every choice of J, for eg., $\mathsf{PVAL}(J, \vec{v})$ is a bad error correcting code when J consists of t copies of the same point.

The argument for completeness trivially holds from the guarantees of Π_{NC} and definition of an LDE of X, since in this case $X \in \mathsf{PVAL}((J, I), (\vec{v}, X|_I))$. We end with a quick note on the complexity of the distribution-free IPP. The query complexity of $O(1/\varepsilon)$ is the same as that of the uniform IPP by [43], and the communication complexity is the sum of the number of bits used to send the $O(1/\varepsilon)$ samples in I in addition to the communication by the uniform IPP, which is $\tilde{O}(\varepsilon n)$. Overall the communication complexity is $\tilde{O}\left(\frac{1}{\varepsilon} + \varepsilon \cdot n\right)$ which matches that in [43] (up to poly-logarithmic factors) whenever $\varepsilon \geq 1/\sqrt{n}$.

1.3.2 Proof outlines of Theorems 3 and 4

Next, we describe the proof techniques of Theorems 3 and 4 that construct IPPs for NC over smooth distributions and product distributions, matching the complexities of [44] for every value of ε. This improves over the communication complexity of the distribution-free IPP in Theorem 1 when $\varepsilon \ll 1/\sqrt{n}$ (with roughly the same query complexity). We follow the general strategy by [44] and the main technical challenges arise during the analysis with respect to the new promise on the soundness of an IPP for PVAL. We assume some familiarity with the uniform IPP construction by [44] for this section.

Uniform IPP for $\mathsf{PVAL}(J, \vec{v})$

We start with a summary of the Uniform IPP from [44]. Let the input $X \in [k]^m$, for $k = \log n$ and $n = k^m$. Further, let $|J| = t$.

[44] use a divide and conquer approach, by decomposing the t claims about X into new claims for each individual row instance $X_i \in \mathcal{F}^{k^{m-1}}$, for every $i \in [k]$. In more detail, let $J = (J_1, J_2)$, where the first component $J_1 \subset \mathcal{F}$ and $J_2 \subset \mathcal{F}^{m-1}$. The prover sends the matrix $Y' \in \mathcal{F}^{k \times t}$, where each row Y'_i is the purported set of evaluations of the $(m-1)$-variate LDE (of individual degree $k-1$) of X_i on J_2. By the definition of an m-variate LDE on X, the prover cannot lie about the consistency of Y' with $\vec{v}$, since for each $(a, b) \in J$ (where $b \in J_2$), the verifier can easily check if the univariate LDE of $Y'[\cdot, b]$ (the b^{th} column of Y) evaluated on the coordinate a equals $\vec{v}[(a, b)]$ (see Figure 2).

Thus, the initial PVAL instance is now reduced to k instances $X_i \in \mathcal{F}^{k^{m-1}}$ for $\{\mathsf{PVAL}(J_2, Y'_i)\}$. A natural idea now is for the verifier to send a random vector $z \in \mathcal{F}^k$ to the prover, and ask it back for a "folded" version $X' \in \mathcal{F}^{k^{m-1}}$, that is purported to be $z \cdot X$.[10] Now, the IPP could recurse on a *single input* $X' \in \mathcal{F}^{k^{m-1}}$ that has shrunk in size by a factor of k, to the problem $\mathsf{PVAL}(J_2, z \cdot Y')$. Completeness easily holds, since if X belonged to $\mathsf{PVAL}(J, \vec{v})$, then the honest prover will just send the "true" $Y' \in \mathcal{F}^{k \times t}$ and the verifier checks always pass.

Uniform Distance Preservation Lemma

However showing soundness is not straightforward. Suppose that X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ under the uniform distribution. It turns out that the malicious prover has cheated in at least one row of the purported matrix Y' (if not, since X is not in PVAL, there would be at least one column in Y' which would be inconsistent with the corresponding value in $\vec{v}$ and the verifier would catch the prover in the checks made above).

[10] The dot product $z \cdot X \in \mathcal{F}^{k^{m-1}}$ between $z \in \mathcal{F}^k$ and a matrix $X \in \mathcal{F}^{k \times k^{m-1}}$ is given by $\sum_{i=1}^{k} z_i X_i$.

For any row $X_i \in \mathcal{F}^{k^{m-1}}$ that is a lower-dimensional input instance, let ε_i be the distance between X_i and $\mathsf{PVAL}(J_2, Y'_i)$. To ensure that the verifier catches the cheating prover, the folded instance X' also needs to be reasonably far from PVAL on a lower dimension at the end of a recursive step. In order to capture this, [44] (implicitly) use a *uniform distance preservation lemma*, which states that if X is ε-far from $\mathsf{PVAL}(J, \vec{v})$, then $\sum_{i=1}^{k} \varepsilon_i > k\varepsilon$.

Using the uniform distance preservation lemma, [44] observe that if the prover ended up cheating (roughly) uniformly across all rows in Y', then any row X_i would be roughly ε-far from $PVAL(J_2, z \cdot Y'_i)$, and the IPP would recurse by picking a single row at random. However, the prover could have cheated across multiple rows of Y'_i and the verifier does not know these rows. To accommodate this, the verifier considers $\log(k)$ many random foldings of X, where the Hamming weight of the vectors z used to fold X, range across 1 to k (in powers of 2). In particular, this results in $O(\log(\log(n)))$ recursive instances in $\mathcal{F}^{k^{m-1}}$. Crucially, they use the uniform distance preservation lemma to generalise the intuition above and show that for at least one of these folded instances, the distance is roughly preserved. Moreover, for such a folded instance, the product of the new distance and the effective query complexity (the number of queries on X to compute the value at any index in $z \cdot X$) is $O(1/\varepsilon)$, along with small but super-constant multiplicative factors.

The IPP continues to recursively fold the instance dimension-wise by the above process, until the size of each final folded instance becomes $\tilde{O}(\varepsilon n)$, which happens after $\Omega(\log(n)/\log(\log(n)))$ steps. In such a case, the prover directly sends each final instance. Since there exists an instance $\tilde{X}^j$ at each level of recursion for which distance is preserved, there exists a final folded instance $\tilde{X}$, such that the verifier catches a cheating prover by uniformly *sampling* a few coordinates of $\tilde{X}$. Moreover, since the product of the distance and effective query complexities for each $\tilde{X}^j$ are roughly maintained to be small at each step of the recursion, making $O(1/\varepsilon^{1+o(1)})$ many queries to $\tilde{X}$ is sufficient to catch the cheating prover (since the total number of recursive instances after the stated number of steps is roughly $n^{o(1)} = 1/\varepsilon^{o(1)}$). The communication complexity is simply the number of bits used to send all the final folded instances, in addition to sending the matrices Y' of size $k \times t$, and thus is $\tilde{O}(\varepsilon^{1-o(1)} n)$.

IPPs for NC under specific distribution families

We now highlight some key ideas which help us construct IPPs over large distribution families like smooth distributions and product distributions. To begin with, on any input $X \in \{0,1\}^{k^m}$, we first reduce L to PVAL using Π_{NC}. Recall that in the distribution-free setting, Π_{NC} outputs $(J, \vec{v})$, such that for the soundness promise, with high probability X cannot be ε-close to $\mathsf{PVAL}(J, \vec{v})$ along both $\mathcal{U}$ and the unknown distribution from the given family, $\mathcal{D}$. In other words, X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ along at least one of $\mathcal{U}$ or $\mathcal{D}$. Building on this observation, we design IPPs for $\mathsf{PVAL}(J, \vec{v})$ over these distribution families, using an intricate case analysis of the soundness condition.

In more detail, if X is ε-far from $\mathsf{PVAL}(J, \vec{v})$ under the uniform distribution, then we can directly use the uniform distance preservation lemma to catch a malicious prover as seen previously in the uniform IPP. If not, suppose that $d_{\mathcal{D}}(X, \mathsf{PVAL}(J, \vec{v})) > \varepsilon$. Next, we briefly describe the soundness analysis, using *specific distance preservation lemmas* for smooth distributions and product distributions. Given this, we build on the strategy of the uniform IPP above to construct an IPP for $\mathsf{PVAL}(J, \vec{v})$ over these distribution families, with the main technical work being that of simultaneously incorporating both the uniform and the respective distance preservation lemmas into the soundness analysis, across the recursive levels.

ρ-dispersed distributions

Recall that ρ-dispersed distributions over $[k]^m$ capture the smoothness of a distribution, by requiring that the probability mass on any element is bounded by ρ times the average mass on any of its neighbours. Adopting similar notation as above, let $\hat{\mathcal{D}}$ be the marginal distribution of $\mathcal{D}$ over $[k]^{m-1}$.

For any row $X_i \in \mathcal{F}^{k^{m-1}}$ that is a lower-dimensional input instance, let ε_i be the distance between X_i and $\mathsf{PVAL}(J_2, Y_i')$ over $\hat{\mathcal{D}}$. Here, we show a distance preservation lemma for ρ-dispersed distributions, such that for any distribution $\mathcal{D}$ that is ρ-dispersed, $\sum_{i=1}^{k} \varepsilon_i > (k\varepsilon)/\rho$.[11] The idea behind proving this is not obvious immediately; while ε_i measures the distance along marginal distributions, ε is the distance from each element of $\mathsf{PVAL}(J, \vec{v})$ over $\mathcal{D}$ (which could be a joint distribution). However, we crucially use properties about ρ-dispersed distributions to prove this distance preservation lemma.

Using the strategy described earlier, we get an IPP for NC over ρ-dispersed distributions, having query and sample complexities $\frac{\rho^{\log(1/\varepsilon)/\log\log(n)}}{\varepsilon^{1+o(1)}}$, while keeping communication complexity the same. In particular, for $\rho = k^{o(1)}$, the query complexity is $1/\varepsilon^{1+o(1)}$ and matches that of the uniform IPP for all $\varepsilon > 0$.

Product distributions

Let $\mathcal{D}$ be an m-product distribution defined as $\mathcal{D} = \mathcal{D}_1 \times \ldots \mathcal{D}_m$ over $[k]^m$, where $k = \log(n)$, and each $\mathcal{D}_j$ is an independent distribution supported on $[k]$. In particular, $\mathcal{D}(i_1, \ldots, i_m)$ is defined as $\prod_{j=1}^{m} \mathcal{D}_j(i_j)$.

Our main approach here to construct IPPs over such distributions, is to first *learn* the underlying distribution and then use this as an aid to obtain near-optimal complexity parameters. For more context, consider the following k-dispersed distribution $\mathcal{D}$ over $[k]^m$, that is supported on the first row of the first dimension, i.e, exactly on the set of elements of the form $(1, i_2, \ldots, i_m)$ for every $(i_2, \ldots, i_m) \in [k]^{m-1}$.[12] We see that the IPP over k-dispersed distributions has query complexity $O(1/\varepsilon^2)$. However, if the verifier "learns" beforehand that $\mathcal{D}$ is only supported on the first row, then it can focus its attention on a smaller instance in $\mathcal{F}^{k^{m-1}}$ and potentially obtain much better query complexity, if $\mathcal{D}$ conditioned on the first row is ρ-dispersed, for a small ρ.

Our main technical idea here is to show a *learning-augmented* distance preservation lemma for product distributions. Let ε_i be the distance between X_i and $\mathsf{PVAL}(J_2, Y_i')$ over $\hat{\mathcal{D}} = \mathcal{D}_2 \times \cdots \times \mathcal{D}_m$. Based on an alternative analysis to that of ρ-dispersed distributions, we prove that for any product distribution $\mathcal{D}$, $\sum_{i=1}^{k} \varepsilon_i > C\varepsilon$, for $C > 1$ that *only depends on* $\mathcal{D}_1$. Using this key insight, if the verifier "transformed" $\mathcal{D}_1$ into the uniform distribution over $[a_0 \cdot k]$, where $a_0 \geq 1$ is a small constant, then we get a similar expression as the uniform distance preservation lemma, i.e., $C = O(k)$, despite still measuring distance according to $\hat{\mathcal{D}}$ for the lower dimensional instances.[13]

We briefly highlight the sequence of tools used to implement the latter idea. The verifier learns the probability vector of $\mathcal{D}_1$, into an approximation $\mathcal{P}_1$, using the *parallel set lower bound protocol* [9] which requires white-box access to $\mathcal{D}_1$. Following this, it runs a

[11] Note that the uniform distribution is a 1-dispersed distribution and we thus generalise the uniform distance preservation lemma.

[12] Intuitively, for any $i_2, \ldots, i_m \in [k]^{m-1}$, $\mathcal{D}(1, i_2, \ldots, i_m)$ is the only element in the set $\{\ell, i_2, \ldots, i_m\}_{\ell \in [k]}$ with a non-zero probability mass and thus is k-times the average of the probability mass on its neighbourhood.

[13] For consistency, $a_0 = 1$, when $\mathcal{D}_1$ is just $\mathcal{U}_k$.

"granularising" algorithm taking $\mathcal{P}_1$ as input, that outputs the probability vector of a new $8k$-granular distribution $\mathcal{E}_1$ over $[k+1]$ (i.e., for every i, $\mathcal{E}_1(i)$ is $b_i/8k$), such that in the soundness case, the distance of the input over $\mathcal{E}_1$ is still ε (up to constant factors). Finally, this granularity set is used to "extend" X into a new input instance $X' \in \{0,1\}^{8k \times k^{m-1}}$, by making copies of each row according to it's granularity, and we can thus, equivalently consider the underlying row distribution as the uniform distribution over $[8k]$. The last two steps build on ideas from [24] for testing unknown distributions, while our focus is on the setting of testing with an implicit input.

1.4 Related Work

Proofs of Proximity for Distributions

In a related model, [14, 35] study proofs of proximity for *testing distributions.* In their setting, for a fixed property Π of distributions, the verifier receives samples from an unknown distribution $\mathcal{D}$, and interacts with the prover to decide whether $\mathcal{D} \in \Pi$ or $\mathcal{D}$ is ε-far from any distribution in Π along the total variation distance. While there are superficial similarities to our model regarding the use of sample oracle, we focus on testing properties (or languages) of strings, where the distribution oracle only provides a means of accessing the input string. In addition, the verifier also has oracle access to the input instance and the distance for the NO instance is measured with respect to the underlying distribution.

Sample-based IPPs

Another related model is that of Sample-based IPPs [20], where the verifier can *only* access the input through an oracle that provides labeled samples over the uniform distribution. They show that any language in logspace-uniform NC has an SIPP with $\tilde{O}(\sqrt{n})$ sample and communication complexities, by in fact constructing a reduction protocol from an SIPP to the query-based IPP by [44]. Our model is more general conceptually, since any protocol in our model needs to be able to test for a language given access to labeled samples over any unknown distribution. On the other hand, to aid with this generality, we also provide the verifier with the more powerful oracle access to the input, which SIPPs do not.

That being said, we can use the uniform SIPP by [20] within the proof of Theorem 1 (instead of the query-based IPP by [43]) to obtain a distribution-free SIPP for NC where the verifier only accesses the input through labeled samples over $\mathcal{U}$ and the unknown distribution $\mathcal{D}$, for any $\varepsilon \geq \tau/n$.[14] It is unclear whether we can construct distribution-free SIPPs for general values of ε (even over smooth or product distributions) that match the complexities of the uniform IPPs and we leave it as future work.

Interactive Proofs for Agnostic Learning

[29] study the setting of verifying PAC-learners. There, the verifier has sampling access to an unknown distribution $\mathcal{D}$ over labeled examples of the form (i, x_i), where $i \sim \mathcal{D}$ and x is the underlying input. It's goal is to verify whether a hypothesis $h : \{0,1\}^{\log(n)} \to \{0,1\}$ given by the prover from a fixed hypothesis class, is the best approximation of $\mathcal{D}$. From the property testing perspective, the prover wants to convince the verifier that $\mathcal{D}'$ has the property that every hypothesis in the class has error larger than ε over $\mathcal{D}$, for some $\varepsilon > 0$ (i.e., the best possible approximation of $\mathcal{D}$ by the hypothesis class is at least ε).

[14] The uniform SIPP by [20] has communication complexity $\tilde{O}\left(\frac{n}{\tau} + \frac{1}{\varepsilon}\right)$ (for tradeoff $\tau \leq \sqrt{n}$), and using this still gives us the same communication complexity as the query-based distribution-free IPP from Theorem 1.

Similar to the setting of SIPPs, their scenario focuses on the case where the verifier only has access to x via a labeled sample oracle, over an unknown distribution. Furthermore, they focus on testing specific properties pertaining to machine learning, such as closeness to an underlying hypothesis class, with the hope of getting very low sample complexity (with respect to the VC dimension of the hypothesis class). In contrast, we deal with verification of general classes of properties, and in some cases the sample and query complexities are both $\tilde{O}(\sqrt{n})$.

References

1 Hugo Aaronson, Tom Gur, Ninad Rajgopal, and Ron Rothblum. Distribution-free proofs of proximity. *Electron. Colloquium Comput. Complex.*, TR23-118, 2023. arXiv:TR23-118.

2 Vipul Arora, Arnab Bhattacharyya, Noah Fleming, Esty Kelman, and Yuichi Yoshida. Low degree testing over the reals. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 738–792. SIAM, 2023.

3 Aleksandrs Belovs. Quantum algorithm for distribution-free junta testing. In René van Bevern and Gregory Kucherov, editors, *Computer Science – Theory and Applications – 14th International Computer Science Symposium in Russia, CSR 2019, Novosibirsk, Russia, July 1-5, 2019, Proceedings*, volume 11532 of *Lecture Notes in Computer Science*, pages 50–59. Springer, 2019. doi:10.1007/978-3-030-19955-5_5.

4 Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon interactive oracle proofs of proximity. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 14:1–14:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ICALP.2018.14.

5 Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. Robust PCPs of proximity, shorter PCPs and applications to coding. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 1–10, 2004.

6 Itay Berman, Ron D. Rothblum, and Vinod Vaikuntanathan. Zero-knowledge proofs of proximity. In Anna R. Karlin, editor, *9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA*, volume 94 of *LIPIcs*, pages 19:1–19:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ITCS.2018.19.

7 Arnab Bhattacharyya and Yuichi Yoshida. *Property Testing – Problems and Techniques.* Springer, 2022. doi:10.1007/978-981-16-8622-1.

8 Eric Blais, Renato Pinto Jr Ferreira, and Nathaniel Harms. VC dimension and distribution-free sample-based testing. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 504–517, 2021.

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

10 Sarah Bordage, Mathieu Lhotel, Jade Nardi, and Hugues Randriam. Interactive oracle proofs of proximity to algebraic geometry codes. In Shachar Lovett, editor, *37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA*, volume 234 of *LIPIcs*, pages 30:1–30:45. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.CCC.2022.30.

11 Nader H. Bshouty. Almost optimal distribution-free junta testing. In Amir Shpilka, editor, *34th Computational Complexity Conference, CCC 2019, July 18-20, 2019, New Brunswick, NJ, USA*, volume 137 of *LIPIcs*, pages 2:1–2:13. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.CCC.2019.2.

12 Xi Chen and Shyamal Patel. Distribution-free testing for halfspaces (almost) requires pac learning. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1715–1743. SIAM, 2022.

13 Xi Chen and Jinyu Xie. Tight bounds for the distribution-free testing of monotone conjunctions. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 54–71. SIAM, 2016. doi:10.1137/1.9781611974331.ch5.

14 Alessandro Chiesa and Tom Gur. Proofs of proximity for distribution testing. In Anna R. Karlin, editor, *9th Innovations in Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA, USA*, volume 94 of *LIPIcs*, pages 53:1–53:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ITCS.2018.53.

15 Marcel Dall'Agnol, Tom Gur, Subhayan Roy Moulik, and Justin Thaler. Quantum proofs of proximity. *Quantum*, 6:834, 2022.

16 Funda Ergün, Ravi Kumar, and Ronitt Rubinfeld. Fast approximate probabilistically checkable proofs. *Inf. Comput.*, 189(2):135–159, 2004. doi:10.1016/j.ic.2003.09.005.

17 Eldar Fischer, Yonatan Goldhirsh, and Oded Lachish. Partial tests, universal tests and decomposability. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 483–500, 2014.

18 Noah Fleming and Yuichi Yoshida. Distribution-free testing of linear functions on $\mathbb{R}^n$. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA*, volume 151 of *LIPIcs*, pages 22:1–22:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ITCS.2020.22.

19 Dana Glasner and Rocco A. Servedio. Distribution-free testing lower bounds for basic boolean functions. In Moses Charikar, Klaus Jansen, Omer Reingold, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 10th International Workshop, APPROX 2007, and 11th International Workshop, RANDOM 2007, Princeton, NJ, USA, August 20-22, 2007, Proceedings*, volume 4627 of *Lecture Notes in Computer Science*, pages 494–508. Springer, 2007. doi:10.1007/978-3-540-74208-1_36.

20 Guy Goldberg and Guy N Rothblum. Sample-based proofs of proximity. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

21 Oded Goldreich. *Introduction to property testing.* Cambridge University Press, 2017.

22 Oded Goldreich. Testing bipartitness in an augmented VDF bounded-degree graph model. *CoRR*, abs/1905.03070, 2019. arXiv:1905.03070.

23 Oded Goldreich. Testing graphs in vertex-distribution-free models. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 527–534, 2019.

24 Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. In Oded Goldreich, editor, *Computational Complexity and Property Testing – On the Interplay Between Randomness and Computation*, volume 12050 of *Lecture Notes in Computer Science*, pages 152–172. Springer, 2020. doi:10.1007/978-3-030-43662-9_10.

25 Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. doi:10.1145/285055.285060.

26 Oded Goldreich and Tom Gur. Universal locally verifiable codes and 3-round interactive proofs of proximity for CSP. *Theoretical computer science*, 878:83–101, 2021.

27 Oded Goldreich and Salil P Vadhan. On the complexity of computational problems regarding distributions. *Studies in Complexity and Cryptography*, 6650:390–405, 2011.

28 Shafi Goldwasser, Yael Tauman Kalai, and Guy N Rothblum. Delegating computation: interactive proofs for muggles. *Journal of the ACM (JACM)*, 62(4):1–64, 2015.

29 Shafi Goldwasser, Guy N Rothblum, Jonathan Shafer, and Amir Yehudayoff. Interactive proofs for verifying machine learning. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

30 Tom Gur. *On locally verifiable proofs of proximity.* PhD thesis, The Weizmann Institute of Science (Israel), 2017.

31 Tom Gur, Govind Ramnarayan, and Ron Rothblum. Relaxed locally correctable codes. *Theory of Computing*, 16(1):1–68, 2020.

32 Tom Gur and Ron D. Rothblum. Non-interactive proofs of proximity. *Comput. Complex.*, 27(1):99–207, 2018. doi:10.1007/s00037-016-0136-9.

33 Shirley Halevy and Eyal Kushilevitz. Distribution-free property-testing. *SIAM J. Comput.*, 37(4):1107–1138, 2007. doi:10.1137/050645804.

34 Shirley Halevy and Eyal Kushilevitz. Distribution-free connectivity testing for sparse graphs. *Algorithmica*, 51:24–48, 2008.

35 Tal Herman and Guy N Rothblum. Verifying the unseen: interactive proofs for label-invariant distribution properties. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1208–1219, 2022.

36 Yael Tauman Kalai and Ron D Rothblum. Arguments of proximity. In *Advances in Cryptology–CRYPTO 2015: 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II*, pages 422–442. Springer, 2015.

37 Zhengyang Liu, Xi Chen, Rocco A Servedio, Ying Sheng, and Jinyu Xie. Distribution-free junta testing. *ACM Transactions on Algorithms (TALG)*, 15(1):1–23, 2018.

38 Zhengyang Liu, Xi Chen, Rocco A. Servedio, Ying Sheng, and Jinyu Xie. Distribution-free junta testing. *ACM Trans. Algorithms*, 15(1):1:1–1:23, 2019. doi:10.1145/3264434.

39 Omer Reingold, Guy N Rothblum, and Ron D Rothblum. Efficient batch verification for UP. In *33rd Computational Complexity Conference (CCC 2018)*. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018.

40 Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. Constant-round interactive proofs for delegating computation. *SIAM J. Comput.*, 50(3), 2021. doi:10.1137/16M1096773.

41 Dana Ron and Asaf Rosin. Optimal distribution-free sample-based testing of subsequence-freeness with one-sided error. *ACM Transactions on Computation Theory (TOCT)*, 14(1):1–31, 2022.

42 Noga Ron-Zewi and Ron D Rothblum. Local proofs approaching the witness length. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 846–857. IEEE, 2020.

43 Guy N Rothblum and Ron D Rothblum. Batch verification and proofs of proximity with polylog overhead. In *Theory of Cryptography: 18th International Conference, TCC 2020, Durham, NC, USA, November 16–19, 2020, Proceedings, Part II*, pages 108–138. Springer, 2020.

44 Guy N. Rothblum, Salil P. Vadhan, and Avi Wigderson. Interactive proofs of proximity: delegating computation in sublinear time. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 793–802. ACM, 2013. doi:10.1145/2488608.2488709.

45 Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM J. Comput.*, 25(2):252–271, 1996. doi:10.1137/S0097539793255151.

46 Amit Sahai and Salil P. Vadhan. Manipulating statistical difference. In Panos M. Pardalos, Sanguthevar Rajasekaran, and José Rolim, editors, *Randomization Methods in Algorithm Design, Proceedings of a DIMACS Workshop, Princeton, New Jersey, USA, December 12-14, 1997*, volume 43 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 251–270. DIMACS/AMS, 1997. doi:10.1090/dimacs/043/14.

47 Salil P Vadhan. An unconditional study of computational zero knowledge. *SIAM Journal on Computing*, 36(4):1160–1214, 2006.

48 Salil Pravin Vadhan. *A study of statistical zero-knowledge proofs.* PhD thesis, Massachusetts Institute of Technology, 1999.

49 Leslie G. Valiant. A theory of the learnable. *Commun. ACM*, 27(11):1134–1142, 1984. doi:10.1145/1968.1972.