Electronic Colloquium on Computational Complexity, Report No. 105 (2025)

Location-Invariant Properties of Functions versus Properties of Distributions: United in Testing but Separated in Verification

Oded Goldreich Guy N. Rothblum

July 29, 2025

Abstract

A property of functions is called location-invariant (or symmetric) if it can be characterized in terms of the frequencies in which each value occurs in the function, regardless of the locations in which each value occurs. It is known that the (query) complexity of testing location-invariant properties of functions is closely related to the (sample) complexity of testing the (corresponding properties of the) corresponding distributions.

The main message of the current work is that this close relationship is not maintained in the context of verification. This holds both when considering verification by general interactive proofs of proximity (i.e., IPPs) and when restricting attention to doubly-sublinear IPPs (ds-IPPs). Alternatively, one may view this work as a subsequent step in the study of doubly-sublinear IPPs (of properties of functions), where we say that an IPP is doubly-sublinear if (1) the query complexity of the verifier is sublinear in the query complexity of testing the property, and (2) the query complexity of the honest prover is sublinear in the query complexity of learning a function in the property.

Specifically, we present doubly-sublinear IPPs for several natural location-invariant properties. Our results include:

- We present doubly-sublinear IPPs for the set of functions from $[m]$ to $[n]$ in which each value occurs m/n times: For every $\alpha \in (0, 0.5)$, the query complexity of the verifier is $O(n^{0.5-\alpha})$, and the query complexity of the honest prover is $\widetilde{O}(n^{0.5+\alpha}/\epsilon^2)$.
- We present doubly-sublinear IPPs for the set of functions from $[m]$ to $[n]$ in which each value occurs either m/k times or not at all: For every $\alpha \in (0, 1/3)$, the query complexity of the verifier is $\mathrm{poly}(1/\epsilon) \cdot k^{(2/3)-2\alpha}$, and the query complexity of the prover is $\mathrm{poly}(1/\epsilon) \cdot \widetilde{O}(k^{(2/3)+\alpha})$.

In contrast, in both cases, it is known that the corresponding properties of distributions have no doubly-sublinear IPP (see Herman and Rothblum, 2025). Actually, the first property of distributions (i.e., uniformity over $[n]$) does not even have an IPP in which the verifier uses $o(n^{1/2})$ samples, regardless of other complexity measures.

Keywords: Property Testing, distribution testing, interactive proofs of proximity, doubly-efficient interactive proofs, doubly-sublinear interactive proofs of proximity.

ISSN 1433-8092

Contents

1 Introduction

A generic description of property testing says that this area studies probabilistic algorithms of sub-linear complexity for deciding whether a given object has a predetermined property or is far from any object having this property. This description avoids the question of what these objects are and how one obtains partial information on them in sub-linear complexity.

The two common answers correspond to two sub-areas of property testing. The arguably main area (which emerged in [8, 29, 15] and is surveyed in [13, Chap. 1–10]) views the tested objects as functions and postulates that partial information is obtained by query access to these functions. In contrast, the sub-area associated with the name *distribution testing* (which emerged in [7, 6] and is surveyed in [13, Chap. 11]), views the tested objects as distributions and postulates that partial information is obtained by samples (of the distribution being tested). The difference between these types of objects is reflected in different orientations of the two areas: the first area is more oriented towards algorithms and complexity, whereas the second area is more oriented towards statistics.

The gap between these two areas is bridged in the study of *testing location-invariant properties of functions* (defined in Section 1.1). These properties can be characterized in terms of the frequencies in which each value occurs in the function, regardless of the locations in which each value occurs. Hence, the correspondence between such properties of functions and properties of distributions is evident: We are actually looking at the probability that a value occurs in a random location (in the function).

As shown in [20, Sec. 6], the (query) complexity of testing location-invariant properties of functions is closely related to the (sample) complexity of testing the (corresponding properties of the) corresponding distributions. The main message of the current work is that this close relation is not maintained in the context of verification. This holds both when considering verification by general interactive proofs of proximity (i.e., IPPs) and when restricting attention to doubly-sublinear IPPs (ds-IPPs). Alternatively, one may view this work as a subsequent step in the study of doubly-sublinear IPPs (of properties of functions).

In order to state our actual results, we first present and discuss the relevant definitions: Location-invariant properties are defined in Section 1.1, whereas IPPs and doubly-sublinear IPPs are defined in Section 1.2. We assuming familiarity with the standard definitions of property testing and distribution testing, reviewed in Appendix A.1.

1.1 Location-invariant properties of functions

We say that a property of functions is location invariant if it is preserved under permuting the domain of the function; that is, the function $f : [m] \to [n]$ is in the property if and only if for every permutation $\pi : [m] \to [m]$ the function $f \circ \pi$ is in the property, where $(f \circ \pi)(i) = f(\pi(i))$ for every $i \in [m]$. (Indeed, we let $[m]$ rather than $[n]$ denote the domain of the function in order to relate such functions to distributions over $[n]$.)

Definition 1.1 (location invariant properties): *Let $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ be a property of functions such that $\Pi_{m,n}$ is a set of functions from $[m]$ to $[n]$. We say that Π is* location invariant *if for every $n, m \in \mathbb{N}$, every $f : [m] \to [n]$, and every permutation $\pi : [m] \to [m]$ it holds that $f \in \Pi_{m,n}$ if and only if $f \circ \pi \in \Pi_{m,n}$, where $(f \circ \pi)(i) = f(\pi(i))$ for every $i \in [m]$.*

In other words, a location invariant property is determined in terms of the frequencies of the different values of functions; that is, a location invariant property $\Pi = \bigcup_{m,n} \Pi_{m,n}$ is determined by a set of frequency functions $\Pi' = \bigcup_n \Pi'_n$; such that $f \in \Pi_{m,n}$ if and only if f' such that $f'(v) = |\{i \in [m] : f(i) = v\}|/m$ is in Π'_n.

Testing location-invariant properties was considered by Goldreich and Ron in [20, Sec. 6] (under the name "symmetric properties"), where it was related to testing properties of distributions. Specifically, a function $f : [m] \to [n]$ is related to the distribution $\mathcal{D}$ such that $\mathcal{D}(v) = |\{i \in [m] : f(i) = v\}|/m$, where $[m]$ is the underlying probability space of $\mathcal{D}$. Likewise, a distribution $\mathcal{D}$ over $[n]$ having an underlying probability space $[m]$ is related to the set of functions $f : [m] \to [n]$ such that $|\{i \in [m] : f(i) = v\}| = m \cdot \mathcal{D}(v)$. Indeed, the distribution $\mathcal{D}$ reflects the frequency function of f.

The difference between testing a location-invariant property and testing the corresponding property of distributions is that in the former case we may make arbitrary queries to the function whereas in the latter case we only obtain samples of the distribution, which correspond to the function values at uniformly and independently distributed locations (in the function's domain; i.e., in $[m]$)).[1] This difference has two aspects:

1. Making queries to the function f versus obtaining random location-value pairs (of the form $(i, f(i))$, where i is uniformly distributed in $[m]$).

   However, as shown in [20, Thm. 6.1], this difference (between general testers and so called sample-based testers) does not matter when testing location-invariant properties.

2. Obtaining random location-value pairs (of the form $(i, f(i))$) versus obtaining random values (i.e., $f(i)$) only.

   As shown in [20, Thm. 6.4], under some natural conditions (i.e., m being sufficiently large), this difference does not matter too much either.

Jumping ahead, we mention that in contrast to the foregoing, the difference between the two settings does matter in the context of IPPs both for general IPPs and also for doubly-sublinear IPPs. That is, better general IPPs (resp., doubly-sublinear IPPs) are available for location-invariant properties of functions, when compared to what is possible for properties of distributions.

A different type of isomorphism. The definition of location invariant properties (i.e., Definition 1.1) may be viewed as referring to a notion of "isomorphism" between functions such that $f : [m] \to [n]$ and $g : [m] \to [n]$ are "isomorphic" if there exists a permutation $\pi : [m] \to [m]$ such that $g = f \circ \pi$ (i.e., $g(i) = f(\pi(i))$ for every $i \in [m]$). This type of "isomorphism" (also considered in [20, 14])[2] is fundamentally different from the (Boolean) *function isomorphism* studied in [2, 3], where $f : \{0,1\}^\ell \to \{0,1\}$ and $g : \{0,1\}^\ell \to \{0,1\}$ are isomorphic if there exists a permuation $\pi : [\ell] \to [\ell]$ such that $f(x_1, ..., x_\ell) = g(x_{\pi(i)}, ..., x_{\pi(\ell)})$.

1.2 IPPs and ds-IPPs

As stated upfront, our focus is on verification rather than on testing. That is, we seek interactive proofs of proximity (IPPs) in which the verifier's complexity is significantly lower than the complexity of testing. Furthermore, our actual focus is on IPPs in which the complexity of the honest

[1] In terms of (a non-standard model of) testing distributions, querying the function corresponds to querying the underlying probability space (i.e., $[m]$). We stress that such queries are not allowed in the standard distribution testing model.

[2] In [14] testing graph isomorphism (in the bounded-degree graph model) is related to testing "location invariant" properties of sequences over an alphabet that encodes the possible connected components in the graph. Needless to say, this relationship is useful only for graphs with small connected components.

prover strategy is significantly lower than the complexity of learning. We start by recalling the relevant definitions.

The notion of an interactive proof of proximity is a hybrid of the notions of property testing and interactive proof systems. The basic mind-frame – of approximate decision and sub-linear complexity – is inherited from the former, whereas the actual setting – of verification via an interaction between a powerful but untrusted prover and a probabilistic verifier – is taken from the latter. In other words, a verifier is a tester that is assisted by interaction with an untrusted prover. In the context of properties of functions, which is the primary focus of this work, we denote by $(\widetilde{P}, V^f)(x)$ the output of V, on input x and oracle access to f, after interacting with an arbitrary prover $\widetilde{P}$.

Definition 1.2 (verifier for property Π):[3] *Let $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ such that $\Pi_{m,n}$ contains functions of the form $f : [m]\to[n]$. An* interactive verifier of proximity for Π *is an interactive and probabilistic oracle machine, denoted V, that, on input parameters m, n and ϵ and oracle access to a function $f : [m] \to [n]$, interacts with a potential prover, and outputs a binary verdict that satisfies the following two conditions.*

1. Completeness: V accepts inputs in Π: *For every $m, n \in \mathbb{N}$ and $\epsilon > 0$, and for every $f \in \Pi_{m,n}$, there exists a prover strategy P that makes the verifier accept* (w.h.p)*; that is,*

$$\Pr[(P, V^f)(m, n, \epsilon)=1] \geq 2/3.$$

If $\Pr[(P, T^f)(m, n, \epsilon)=1] = 1$ *always holds, then we say that V has* perfect completeness.

2. Soundness: V rejects inputs that are ϵ-far from Π: *For every $m, n \in \mathbb{N}$ and $\epsilon > 0$, and for every $f : [m] \to [n]$ such that $\delta_\Pi(f) > \epsilon$, no prover strategy $\widetilde{P}$ can make the verifier accept* (w.h.p)*; that is,*

$$\Pr[\widetilde{P}, V^f)(m, n, \epsilon)=0] \geq 2/3,$$

where $\delta(f, g) \overset{\text{def}}{=} |\{i\in[m] : f(i)\neq g(i)\}|/m$ *and* $\delta_\Pi(f) \overset{\text{def}}{=} \min_{g\in\Pi_{m,n}} \{\delta(f, g)\}$.

The query complexity of V *is a function* (of m, n and ϵ) *that specifies the number of queries made by V on input parameters m, n and ϵ, when given oracle access to any function $f : [m]\to[n]$.*

An alternative formulation specifies the (honest) prover strategy that is used in the completeness condition. This is called for when wishing to discuss the complexity of such honest prover strategies. Indeed, verifier of proximity that admit an efficient proving strategy (in case $f \in \Pi$) are of natural interest, and are our main focus. In these cases, we provide these prover strategies with oracle access to the input function and consider their query complexity. Hence, we use the following definition.

Definition 1.3 (an IPP for property Π): *For Π and V as in Definition 1.2, we say that (P, V) is an* interactive proof of proximity (IPP) for Π *if P is an interactive oracle machine and Condition 1* (completeness) *holds with P replaced by P^f; that is, for every $m, n \in \mathbb{N}$ and $\epsilon > 0$ and for every $f \in \Pi_{m,n}$, it holds that*

$$\Pr[(P^f, V^f)(m, n, \epsilon)=1] \geq 2/3.$$

[3] Since our focus is on properties of the form $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ (as in Definition 1.1), we provide all machines with two size parameters (i.e., m and n). The definition of a tester can be derived as a special case by considering verifiers that do not interact with the prover.

We stress that V also satisfies Condition 2 (soundness). *The* query complexity of P *is a function that specifies the number of queries made by P on input parameters m, n and ϵ, when given oracle access to any function $f : [m] \to [n]$.*

Analogous definitions are used for interactive proofs of proximity (IPPs) for distributions (cf. Definition A.2). Since we refer to such IPPs for the sake of comparison only, we do not define them here.

We say that (P, V) is a doubly-sublinear IPP (ds-IPP) for Π if (1) the query complexity of V is sublinear in the query complexity of *testing* Π, and (2) the query complexity of P is sublinear in the query complexity of *learning* Π. We stress that this definition is minimal: It only refers to the query complexity of the two parties, and it only requires some advantage over the straightforward case.[4]

1.3 Our results

As stated in Section 1.1, testing a location-invariant property of functions Π is closely related to testing a related property of distributions $\mathcal{D}$, where the distributions in $\mathcal{D}$ reflect the frequencies of the functions in Π. This close relationship is based on the fact that making queries to a function (in Π) is not more beneficial than obtaining random samples of the function's value.

The latter assertion, rigorously stated and proved in [20, Sec. 6], no longer holds in the context of IPPs. In the latter context, the ability to make queries (rather than get samples) is advantageous (even when the property is location-invariant). This is the case because it may be beneficial for the verifier to check the validity of location-value pairs provided by the prover. We stress that such a possibility exists in the setting of properties of functions, but not in the setting of properties of distributions. This is the source of the gap between IPPs for properties of functions and IPPs for corresponding properties of distributions. Let us illustrate this point by considering the property of location-invariant functions that corresponds to the uniform distribution.

Definition 1.4 (the equal-frequencies property): *We say that $f : [m] \to [n]$ is an* equal-frequencies function *if for every $v \in [n]$ it holds that $\#_v(f) \overset{\rm def}{=} |\{i \in [m] : f(i) = v\}| = m/n$. The set of equal-frequencies functions is called the* equal-frequencies property.

We first comment that testing the equal-frequencies property requires $\Omega(\sqrt{n})$ queries. Furthermore, an IPP for uniform distribution (over $[n]$) requires the verifier to make $\Omega(\sqrt{n})$ queries [9]. However, as shown next, the equal-frequencies property has an IPP in which the verifier runs in poly$(1/\epsilon)$-time.

The latter (non-doubly-sublinear) IPP proceeds essentially as follows. Given oracle access to $f : [m] \to [n]$, *the verifier selects $i \in [m]$ uniformly at random, obtains $v \leftarrow f(i)$ and asks the prover to prove to it that $|f^{-1}(v)| = m/n$.* As shown next, such a proof can be carried out in the current context (of functions, but not in the context of distributions).

The crucial observation is that the standard lower and upper bound protocols (see [24, 17] and [11, 1], respectively) are applicable here: the lower bound protocol is applicable because the verifier can check claims of membership in $f^{-1}(v)$, whereas the upper bound protocol is applicable here because the verifier keeps $i \in f^{-1}(v)$ secret. In both protocols, the parties use a random

[4] Evidently, verification is reducible to testing, whereas proving is essentially reducible to (perfectly) learning the function (when considering the query complexity only).

hashing function $h : [m] \to [m/(t \cdot n)]$, where $t = \mathrm{poly}(1/\eta)$, and the prover is asked to answer with $(1 \pm \eta) \cdot t$ indices j such that $f(j) = v$ and $h(j) = 0$.[5] Unfortunately, implementing the prover's strategy essentially requires full knowledge of f.

Our main result is showing that the aforementioned gap between IPPs for location-invariant properties of functions and IPPs for the corresponding properties of distributions occurs also in case of doubly-sublinear IPPs (ds-IPPs). For example, we show that doubly-sublinear IPPs exist for the equal-frequencies property.

Theorem 1.5 (ds-IPP for the equal-frequencies property): *For every $\alpha \in (0, 0.5)$, the equal-frequencies property* (of functions from $[m]$ to $[n]$) *has an IPP in which the query complexity of the verifier is $O(n^{0.5-\alpha})$, the query complexity of the honest prover is $p = \widetilde{O}(n^{0.5+\alpha}/\epsilon^2)$, and the communication complexity is $O(p \cdot \log m)$.*

Recall that this result stands in contrast to the analogous situation wrt ds-IPPs for testing the uniform distribution [9, 26], where IPPs for this property (of distributions) offer no advantage over testers. (Recall that an IPP for testing uniformity over $[n]$ must have sample complexity $\Omega(\sqrt{n})$, regardless of other complexity measures.)

Theorem 1.5 extends to any *fixed-frequencies property*, where for any $\rho : [n] \to [0, 1]$ that sums to 1, we say that $f : [m] \to [n]$ is a ρ-frequencies function if for every $v \in [n]$ it holds that $\#_v(f) = \rho(v) \cdot m$ (i.e., the equal-frequencies property corresponds to the case in which $\rho(v) = 1/n$ for every $v \in [n]$). This extension is shown by carefully using known reductions (see [13, Sec. 11.2.2]). A different generalization refers to the following (locally-invariant) property.

Definition 1.6 (the flat-frequencies property): *We say that $f : [m] \to [n]$ is a* flat-frequencies function *if f is a ρ-frequencies function for a frequency function ρ such that for some $k \in \mathbb{N}$ and every $v \in [n]$ it holds that $\rho(v) \in \{0, 1/k\}$. In this case we say that f is k*-flat*. The set of flat-frequencies functions is called the* flat-frequencies property*.*

Indeed, the equal-frequencies functions corresponds to the special case of flat-frequency in which $k = n$. Note that the flat-frequencies property (of functions) corresponds to generalized uniformity studied in [5, 10]. Recall that testing generalized uniformity has query complexity $\Theta(k^{2/3})$ [5], which cannot be improved by ds-IPPs [26]; that is, obtaining sample complexity $o(k^{2/3})$ for the verifier requires sample complexity $\Omega(k)$ for the prover. In contrast, we show

Theorem 1.7 (ds-IPP for the flat-frequencies property): *For every $\alpha \in (0, 1/3)$ and every $k \in [n]$, the k-flat property* (of functions from $[m]$ to $[n]$) *has an IPP in which the query complexity of the verifier is* $\mathrm{poly}(1/\epsilon) \cdot k^{(2/3)-2\alpha}$*, the query complexity of the prover is $p = \mathrm{poly}(1/\epsilon) \cdot \widetilde{O}(k^{(2/3)+\alpha})$, and the communication complexity is $O(p \cdot \log m)$.*

The foregoing IPP requires the honest prover to have a good approximation of k, which it can obtain by itself (using the tester of [5]).

[5] Recall that if $|f^{-1}(v)| \geq m/n$ (resp., $|f^{-1}(v)| < (1 - 2\eta) \cdot m/n$), then (whp) $|f^{-1}(v) \cap h^{-1}(0)| \geq (1 - \eta) \cdot t$ (resp., $|f^{-1}(v) \cap h^{-1}(0)| < (1 - \eta) \cdot t$). Likewise, if $|f^{-1}(v)| \leq m/n$ (resp., $|f^{-1}(v)| > (1 + 3\eta) \cdot m/n$), then (whp) $|f^{-1}(v) \cap h^{-1}(0)| \leq (1 + \eta) \cdot t$ (resp., $|f^{-1}(v) \cap h^{-1}(0)| > (1 + 2\eta) \cdot t$, which means that with probability $\Omega(\eta)$ the index i is not in the $(1 + \eta) \cdot t$-subset sent by the prover).

Digest: In all our IPPs, the query complexity of the prover as well as the communication complexity is larger than the query complexity of the corresponding tester. Hence, the gain in reducing the query complexity of the verifier comes at the cost of increasing the query complexity of the prover (above the query complexity of the tester). Furthermore, the communication complexity (and so the verifier's running time) is also at the latter level. So the verifier gains in reducing its query complexity at the cost of increasing its running-time (and communication complexity). This trade-off is beneficial in settings in which the query access is more costy than running-time and communication.

IPPs for all locally-invariant properties. Turning back to general IPPs, which are not necessarily doubly-sublinear, we observe that every locally-invariant property has an IPP in which the verifier makes $O(1/\epsilon)$ queries.

Theorem 1.8 (IPP for any locally-invariant property): *Every locally-invariuiant property* $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ *has an IPP in which the query complexity of the verifier is* $O(1/\epsilon)$*, the query complexity of the prover is* $p = O(n/\epsilon^2)$*, and the communication complexity is* $O(p \cdot \log m)$*.*

Recall that the corresponding distribution tester requires $\Omega(n/\log n)$ queries. Hence, the query complexity of the prover in Theorem 1.8 is optimal up to a polylogarithmic (in n) factor.

1.4 Techniques

The common theme in all our doubly-sublinear IPPs is that the verifier delegates the task of querying the function f to the (untrusted) prover. In light of the fact that we aim at verifying location-invariant properties, it suffices to obtain the value of the function at random locations, but the verifier has to make sure that these locations are indeed random and that the prover returns the correct values. Specifically, in all cases, the ds-IPP (for Π) proceeds as follows, where p is the query complexity of the prover and $q \ll p$ is the query complexity of the verifier.

1. The verifier selects p locations, $i_1, ..., i_p \in [m]$, uniformly at random, and sends them to the prover.

2. The prover queries the function $f : [m] \to [n]$ at these p locations, obtains $v_j = f(i_j)$ for each $j \in [p]$, and sends $v_1, ..., v_p$ to the verifier.

3. The verifier subjects $v_1, ..., v_p$ to some check, which is related to some known tester for Π. If the check fails, the verifier rejects. Otherwise, it proceeds to the next step.

4. The verifier selects a random q-subset J of $[p]$ and queries f on i_j for all $j \in J$. It accepts if and only if $f(i_j) = v_j$ for every $j \in J$.

The check performed in Step 3 typically amounts to counting the number of pairwise (and 3-wise) collisions in the sequence $v_1, ..., v_p$. (Indeed, this corresponds to the way uniformity and generalized uniformity are tested; see [13, Sec. 11.2.1] and [5].)

As usual, the main challenge is the soundness analysis. We show that passing the check performed in Step 3 requires cheating on sufficiently many (i.e., $\omega(p/q)$) values, whereas such cheating will be detected in Step 4. Proving this is relatively easy in the special case of Theorem 1.5 in which $m = n$. In this case, the set $\Pi_{m,n}$ of equal-frequencies functions coincides with the set $\texttt{PERM}_n$ of permutations over $[n]$. This special case was outlined by us in [4, Sec. 1.2.3], and a full analysis of it is presented in Section 2.

The analysis of the ds-IPP for $\texttt{PERM}$ is facilitated by the fact that in that case no collisions may exist under a tested function $f \in \texttt{PERM}$. In contrast, in the other cases, we have to approximate the number of (pairwise and 3-wise) collisions, and this is more complicated for arbitrary tested functions f. In fact, we reduce the analysis to the case that no value occurs in f with frequency larger than $1/p$ (or so). In some of these cases, the reduction is explicit, and in others it is implicit. In Section 5 we distill and study the computational problem that underlies these reductions (i.e., IPP for "frequency-bounded" properties).

Evidently, all these IPPs use two messages. While the IPP for $\texttt{PERM}$ has perfect completeness, this is not the case for our other IPPs. The latter deficiency is inherent to the fact that these IPPs are based on estimating various quantities.

Comment: In all cases, the analysis holds also if the p locations selected in Step 1 are selected in an $O(1)$-wise independent manner rather than totally independent of one another. While this reduces the length of the message sent in Step 1, it does not reduce the length of the message sent in Step 2 (nor the time required for checking it in Step 3).

On the proof of Theorem 1.8. Our IPP for any locally-invariant property also follows the foregoing theme. Specifically, the verifier learns the frequency function of the input function $f : [m] \to [n]$ (i.e., $\rho : [n] \to [0,1]$ s.t. $\rho(v) = \#_v(f)/m$ for each v) by delegating the corresponding $O(n/\epsilon^2)$ queries to the prover. The verifier checks whether this frequency function is close to the set of frequency functions underlying the property, and that the $O(n/\epsilon^2)$ values provided by the prover are actually correct (by querying a sub-sample of $O(1/\epsilon)$ points).

1.5 Prior works on which we build

Evidently, we refer to a host of notions that were defined in prior works. In particular, as stated above, property testing emerged in the works of Blum, Luby and Rubinfeld [8], Rubinfeld and Sudan [29], and Goldreich, Goldwasser and Ron [15]. Distribution testing (although mentioned in [15]) emerged in the work of Batu, Fortnow, Fischer, Kumar, Rubinfeld, Smith and White [7, 6].

Interactive proofs of proximity (for functions) were defined by Rothblum, Vadhan, and Wigderson [27], whereas interactive proofs of proximity for distributions were later defined by Chiesa and Gur [9]. Doubly-sublinear IPPs (for functions) were recently defined and studied in [4], whereas doubly-sublinear IPPs for distributions were studied (even more recently) in [26]. (Evidently, IPPs adapt the notion of interactive proofs [23] to the context of approximate decisions, whereas doubly-sublinear IPPs adapt the notion of doubly-efficient interactive proofs [22] to that context.)

In addition, we also refer and use a few prior results. For example, location-invariant properties were previously considered by Goldreich and Ron [20, Sec. 6], in the context of studying "'sample-based" testers. In particular, they showed that testing such properties (of functions) is essentially equivalent testing properties of distributions (i.e., the corresponding "frequencies" distributions). Our IPPs utilize ideas that underlie testing the uniform distribution (see [13, Sec. 11.2.1]), testing equality to fixed distributions (see [13, Sec. 11.2.2]), and testing generalized uniformity [5].

We also reproduce the ds-IPP for the set of permutations over $[n]$, denoted $\texttt{PERM}_n$, presented by us in [4, Sec. 1.2.3], and revise and detail its analysis. This serves as a good warm-up for the rest of our results.

1.6 Organization

As stated above, the doubly-sublinear IPP for PERM, presented in Section 2, is a good warm-up towards the rest of our results. The main results of this work are presented in Sections 3 and 4, which contain proofs of Theorem 1.5 and 1.7, respectively. In Section 5 we distil and study a computational problem that arises in the previous two sections, but reading this section is not essential for the results presented in Sections 3 and 4. Lastly, in Section 6, we present the proof of Theorem 1.8 (asserting IPPs for any locally-invariant property).

2 Warm-up: IPPs for PERM

For $n \in \mathbb{N}$, the set $\texttt{PERM}_n$ consists of all permutations over $[n]$. Indeed, $\texttt{PERM} = \bigcup_n \texttt{PERM}_n$ is a specail case of the equal-frequencies property; this special case is obtained by setting $m = n$.

Here we consider the problem of testing (resp., verifying) whether a function $f : [n] \rightarrow [n]$ is in $\texttt{PERM}_n$. Note that PERM has a tester of complexity $O(\sqrt{n/\epsilon})$, which we outline below, whereas a query lower bound of $\Omega(\sqrt{n})$ follows from [25].[6] Furthermore, PERM has two fundamentally different (1-round) IPPs, which were presented in [25] and [16], respectively. We review them next:

1. In the IPP of [25] the verifier selects uniformly at random a point $r \in [n]$, and sends r to the prover, who is supposed to return its f-preimage; the verifier accepts if and only if the prover's answer is mapped by f to r.

   (The analysis of this IPP relies on the fact that the distance of $f : [n] \rightarrow [n]$ from $\texttt{PERM}_n$ is linearly realted to the number of elements in $[n]$ that have no preimage under f (i.e., $n - |f([n])|$).)

2. In the IPP of [16], the verifier selects uniformly at random a point $r \in [n]$, queries f on it, and sends $y \leftarrow f(r)$ to the prover, who is supposed to return its f-preimage; the verifier accepts if and only if the prover's answer equals r.

   (This IPP, unlike the previous one, utilizes prover-oblivious queries. Its analysis relies on the fact that the distance of $f : [n] \rightarrow [n]$ from $\texttt{PERM}_n$ is linearly related to $\sum_{v \in F([n])}(|f^{-1}(v)| - 1)$.)

In both cases, $f \in \texttt{PERM}_n$ is always accepted, whereas functions that are ϵ-far from PERM are rejected with probability $\Omega(\epsilon)$. (In both cases, the protocol is repeated $O(1/\epsilon)$ times to yield an IPP.)

More importantly, in both cases, the honest prover finds the required f-preimage by querying f on all points (or practically so). In contrast, we seek and obtain a ds-IPP for PERM in which the verifier uses $o(\sqrt{n})$ queries and a honest prover that makes $o(n)$ queries.

We stress that the following result stands in contrast to the analogous situation wrt IPPs for testing the uniform distribution [**?**, 26]. (Recall that an IPP for testing uniformity over $[n]$ must have sample complexity $\Omega(\sqrt{n})$, regardless of other complexity measures.)

[6] See also a direct proof in [30, Apdx A].

Theorem 2.1 (ds-IPP for `PERM`): *For every $\alpha \in (0, 0.5)$, there exists an IPP for* $\texttt{PERM}_n$ *that has a verifier that uses* $O(n/\epsilon)^{0.5-\alpha}$ *queries and an honest prover that uses* $O(n/\epsilon)^{0.5+\alpha}$ *queries. Furthermore, the IPP has perfect completeness.*

The communication and time complexity of both parties is $\widetilde{O}((n/\epsilon)^{0.5+\alpha})$.

Proof: The straightforward tester for `PERM` consists of selecting $q = O(\sqrt{n/\epsilon})$ random points in $[n]$, querying the function $f : [n] \to [n]$ on them, and rejecting if and only if collisions are found (among the f-images).

Evidently, any $f \in \texttt{PERM}$ is accepted with probability 1, whereas (as implicitly shown below) any f that is ϵ-far from `PERM` is rejected with high constant probability. A similar analysis implies that if we select $O(n/\epsilon)^{0.5+\alpha}$ random points, then we expect to see $\Omega(n^{2\alpha})$ collisions and that these collisions involve $\Omega(n^{2\alpha})$ disjoint pairs of points. This leads us to the following protocol.

1. The verifier selects $p = O(n/\epsilon)^{0.5+\alpha}$ (distinct) random points in $[n]$, denoted $i_1, ..., i_p$, and sends them to the prover,
2. The prover queries the input $f : [n] \to [n]$ on these p sample points, and sends the answers $(v_1, ..., v_p) \leftarrow (f(i_1), ..., f(i_p))$ to the verifier.
3. The verifier rejects if it sees a collision (i.e., if $v_j = v_k$ for some $j \neq k$).
4. Otherwise, the verifier sub-samples $q = O(n/\epsilon)^{0.5-\alpha}$ of the original points, queries f on each of these q samples, and accepts if and only if all answers match the prover's answers (i.e., if $f(i_j) = v_j$ for the i_j's it sub-sampled).

Clearly, $f \in \texttt{PERM}$ is always accepted. Turning to the soundness analysis, we fix an arbitrary function f that is ϵ-far from `PERM` and let $C \stackrel{\rm def}{=} \{x \in [n] : |f^{-1}(f(x))| > 1\}$ denote the set of points that form collisions under f. Then, $|C| > \epsilon \cdot n$; actually, $|C| - |f(C)| > \epsilon \cdot n$, because modifying f to a permutaion requires changing its value on all but a single point in $f^{-1}(y)$ for every $y \in f(C)$.

Actually, let use first consider the *special case in which f is 2-to-1 on C*. In this case, C consists of pairs of the form (x, y) such that $f(x) = f(y)$. Each such pair is included in the sample $S = \{i_1, ..., i_p\}$ taken by the verifier with probability $\binom{p}{2}/\binom{n}{2} \approx (p/n)^2$, and so the expected number of pairs included in S is $\approx (p/n)^2 \cdot |C| \geq \epsilon \cdot p^2/n = \Omega((n/\epsilon)^{2\alpha})$. Furthermore, with high probability over the choice of $S \in \binom{[n]}{p}$, there are $\Omega((n/\epsilon)^{2\alpha})$ disjoint pairs of points in S such that the elements in each pair have the same f-image.[7]

Wishing to avoid rejection in Step 2, the prover must cheat on the values of the $\Omega((n/\epsilon)^{2\alpha})$ foregoing points (in S), but (with high probability) at least one of these points will appear in the sub-sample that the verifier selects in Step 4 (since the sub-sample rate is $q/p = O(\epsilon/n)^{2\alpha}$). Hence, in Step 4, when querying the function f on this sub-sample, the verifier detects this cheating and rejects (w.h.p.). Recall however, that this analysis was conducted under the unjustified assumption that f is 2-to-1 on C.

[7] The proof relies on the fact that the relevant events (i.e., pairwise collisions) are sufficiently independent. Specifically, most pairs of events refer to four independent samples and the few pairs of events that refer to three independent samples correspond to three-way collisions and occur with very small probability. See details in Appendix A.2, where the current case corresponds to a graph with vertex set $[n]$ such that x is connected to y if $f(x) = f(y)$. (Hence, this graph consists of $|C|/2$ isolated edges and $n - (|C|/2)$ isolated vertices.)

Turning to the general case, we define a function f' that is 2-to-1 on most of C such that $f(x) \neq f(y)$ implies $f'(x) \neq f'(y)$ (equiv., $f'(x) = f'(y)$ implies $f(x) = f(y)$). Hence, the probability that the verifier rejects f' (when interacting with the worst possible cheating prover) is upper-bounded by the probability that the verifier rejects f (when interacting with the worst possible cheating prover).[8] The theorem follows by using the foregoing analysis (because f' is 2-to-1 on a set C' that contains most of C). The desired function f' is defined as follows.

- For each $x \in [n]$, let $E_x = f^{-1}(f(x)) = \{y \in [n] : f(y) = f(x)\}$.
- For each $x \in C$ (equiv., x s.t. $|E_x| > 1$), partition E_x to pairs, leaving at most one unpaired element (where $|E_x|$ is odd). This yields $t \stackrel{\rm def}{=} \sum_v \lfloor |f^{-1}(v)|/2 \rfloor \geq |C|/3$ pairs.
- Denoting these pairs $(x_1, y_1), ..., (x_t, y_t)$, let f' assign both elements of each pair a distinct value that is different from all values assigned by f (to elements that are not in these pairs). That is, letting $C' \stackrel{\rm def}{=} \{x_i, y_i : i \in [t]\}$, and assuming that $v_1, ..., v_t$ are not in $f([n] \setminus C')$, we let
$$f'(z) \stackrel{\rm def}{=} \begin{cases} v_i & \text{if } z \in \{x_i, y_i\} \\ f(z) & \text{otherwise} \end{cases}$$

Observing that f' is 2-to-1 on C' and that $|C'| \geq 2 \cdot |C|/3 > \epsilon \cdot n/2$, we complete the proof of Theorem 2.1. ■

3 IPPs for fixed-frequency properties

We start with an explicit definition of the properties that we study in the current section.

Definition 3.1 (the fixed-frequencies property): *For $\rho : [n] \to [0, 1]$ such that $\sum_{v \in [n]} \rho(v) = 1$, we say that $f : [m] \to [n]$ is a ρ-frequencies function if for every $v \in [n]$ it holds that $\#_v(f) = \rho(v) \cdot m$, where $\#_v(f) \stackrel{\rm def}{=} |\{i \in [N] : f(i) = v\}|$. The set of ρ-frequencies functions is called the ρ-frequencies property.*

Indeed, equal-frequencies functions (see Definition 1.4) are a special case of ρ-frequencies functions (i.e., $\rho(v) = 1/n$ for every $v \in [n]$). We shall first present doubly-sublinear IPPs for this special case, and then use known reductions of the general case to this special case. These reductions, originally presented in the context of testing distributions, are shown to hold in the current setting (of constructing ds-IPPs for the corresponding location-invariant properties of functions).

Recall that PERM is a special case of the equal-frequencies property (obtained by setting $m = n$). In general, equal-frequencies functions from $[m]$ to $[n]$ are m/n-to-1 (rather than 1-to-1 as in the case of $m = 1$), which makes the construction of IPPs for them more complex.

3.1 IPPs for the equal-frequencies property: Proof of Theorem 1.5

The equal-frequencies property is a special case of the fixed-frequencies property, obtained when considering the uniform frequency; that is, $f : [m] \to [n]$ is an equal-frequencies function if for every $v \in [n]$ it holds that $\#_v(f) = m/n$.

[8] These probabilities are related to $|S| - |f'(S)|$ and $|S| - |f(S)|$, respectively, and the foregoing claim follows by observing that $|f'(S)| \geq |f(S)|$.

Recall that testing the equal-frequencies property requires $\Omega(\sqrt{n})$ queries, but it has an IPP in which the verifier runs in $\text{poly}(1/\epsilon)$-time. The latter (non-doubly-sublinear) IPP proceeds essentially as follows. Given oracle access to $f : [m] \to [n]$, *the verifier selects* $i \in [m]$ *uniformly at random, obtains* $v \leftarrow f(i)$ *and asks the prover to prove to it that* $|f^{-1}(v)| = m/n$.

However, our goal is to present a doubly-subliner IPP for the foregoing property. Specifically, for every constant $\alpha \in (0, 0.5)$, we present an IPP in which the prover makes $p = \widetilde{O}(n^{0.5+\alpha}/\epsilon^2)$ queries whereas the verifier makes $q = O(n^{0.5-\alpha})$ queries. We may assume, w.l.o.g., that $p < m$, since otherwise the prover can retrieve f using p queries.

Construction 3.2 (ds-IPP for the equal-frequencies property): *On input* m, n *and* ϵ, *and oracle access to* $f : [m] \to [n]$, *the proof system proceeds as follows.*

1. *The verifier selects* $i_1, ..., i_p \in [m]$ *uniformly at random, and sends* $(i_1, ..., i_p)$ *to the prover.*
2. *For every* $j \in [p]$, *the prover obtains* $v_j \leftarrow f(i_j)$, *by querying* f, *and sends* $(v_1, ..., v_p)$ *to the verifier.*
3. *The verifier subjects the received* p*-long sequence* $(v_1, ..., v_p)$ *to two checks:*
   (a) *If some value occurs more than* $\log_2 n$ *times in the sequence* (i.e., if $|\{j \in [p] : v_j = v\}| > \log_2 n$ for some $v \in [n]$), *then the verifier rejects.*
   (b) *If the empirical collision probability in the* p*-long sequence exceeds* $(1 + 2\epsilon^2)/n$, *then the verifier rejects. That is, the verifier rejects if* $|\{\{j, k\} \in \binom{[p]}{2} : v_j = v_k\}|$ *is greater than* $\frac{1+2\epsilon^2}{n} \cdot \binom{p}{2}$.
   Recall that the uniform distribution over $[n]$ *has collision probability* $1/n$, *whereas a distribution over* $[n]$ *that is* ϵ*-far from uniform* (over $[n]$) *has collision probability greater than* $(1 + 4\epsilon^2)/n$.

   Otherwise (i.e., in case the verifier did not reject), *the verifier proceeds to the next step.*
4. *The verifier selects uniformly at random a* q*-subset* $J \subset [p]$, *queries* f *on each element of* $\{i_j : j \in J\}$, *and accepts if and only if* $v_j = f(i_j)$ *for every* $j \in J$.

If f is an equal-frequencies function, then, with high probability, the correct sequence $(f(i_1), ..., f(i_p))$ passes both checks of Step 3. Actually, Step 3a is unnecessary, but it simplifies the soundness analysis (i.e., of the case that f is ϵ-far from the property), which is presented next.

In the simplified analysis (which relies on Step 3a), we may assume that f is 0.1ϵ-close to f' such that no element occurs in f' with frequency greater than $2 \cdot \frac{m}{p} \cdot \log_2 n$, because otherwise either Step 3a rejects (whp) or the prover has to cheat on $\Omega(\epsilon)$ of the values (and gets caught (whp) by Step 4). As a mental experiment, we consider an execution with f', which is 0.9ϵ-far from the equal-frequencies property, rather than with f. Note that we can choose f' such that $f'(i) = f'(j)$ implies $f(i) = f(j)$; hence, $|\{\{j, k\} \in \binom{[p]}{2} : f'(i_j) = f'(i_k)\}| \leq |\{\{j, k\} \in \binom{[p]}{2} : f(i_j) = f(i_k)\}|$, which means that if an execution with f' rejects than the corresponding execution with f rejects too.

Recall that if f' is 0.9ϵ-far from uniform, then its collision probability exceeds $(1 + 4 \cdot (0.9\epsilon)^2)/n$, and, with high probability (see Appendix A.2)[9], the true values that correspond to the sample

[9] Analogously to Footnote 7, the current case corresponds to a graph with vertex set $[m]$ such that x is connected to y if $f(x) = f(y)$. In the current case, the maximum degree of this graph is $O((m/p) \cdot \log n)$ whereas the average degree is at least m/n.

(i.e., $(f'(i_1),...,f'(i_p)))$ will have empirical collision probability greater than $(1+3\epsilon^2)/n$; that is, $|\{\{j,k\}\in\binom{[p]}{2}:f'(i_j)=f'(i_k)\}|$ exceeds $\frac{1+3\epsilon^2}{n}\cdot\binom{p}{2}$. To avoid rejection in Step 3b, the prover must avoid $\Delta=\frac{\epsilon^2}{n}\cdot\binom{p}{2}=\Omega(\epsilon^2p^2/n)$ of the collisions, which requires it to cheat on at least $\Delta/\log_2 n$ values (where here we use the bound on frequencies of f' in the sample). Hence, the probability of catching this cheating (in Step 4) is high, since the verifier checks each value with probability $\frac{q}{p}$ (i.e., the expectation is $(q/p)\cdot\Delta/\log_2 n=\Omega(\epsilon^2qp/n\log n)=\omega(1)$ (by the setting of p and q) and there is sufficient concentration (see Footnote 9 and Appendix A.2)).

An alternative analysis (which avoids Step 3a). The alternative analysis (of f that is ϵ-far from the equal-frequencies property) is closer in spirit to the analysis presented in Section 2. Firstly, we say that a value v is heavy (in f) if $\#_v(f)\geq 3\cdot m/n$, and let H denote the set of heavy values. Next, we replace each heavy value v by a set of $\lfloor\frac{\#_v(f)}{2m/n}\rfloor$ auxiliary values, denoted C_v. Next, we define f' such that $f'(i)\in C_{f(i)}$ if $f(i)\in H$ and $f'(i)=f(i)$ otherwise. Moreover, we define f' such that each value in C_v is assigned approximately the same frequency; that is, for every $v\in H$ and $w\in C_v$ it holds that

$$\#_w(f') \;\approx\; \frac{\#_v(f)}{|C_v|} \;=\; \frac{\#_v(f)}{\lfloor\#_v(f)\cdot(2m/n)^{-1}\rfloor} \;\in\; \left[2\cdot\frac{m}{n}\;,\;4\cdot\frac{m}{n}\right) \tag{1}$$

and for every $v\in H$ it holds that

$$\begin{aligned}\sum_{w\in C_v}\left(\#_w(f')-\frac{m}{n}\right) &= \#_v(f)-|C_v|\cdot\frac{m}{n}\\ &\geq \#_v(f)-\frac{\#_v(f)}{2m/n}\cdot\frac{m}{n}\end{aligned}$$

which equals $\frac{\#_v(f)}{2}$. It follows that

$$\begin{aligned}\sum_{w:\#_w(f')>m/n}\left(\#_w(f')-\frac{m}{n}\right) &= \sum_{v\notin H:\#_v(f)>m/n}\left(\#_v(f)-\frac{m}{n}\right)+\sum_{v\in H}\sum_{w\in C_v}\left(\#_w(f')-\frac{m}{n}\right)\\ &\geq \sum_{v\notin H:\#_v(f)>m/n}\left(\#_v(f)-\frac{m}{n}\right)+\frac{1}{2}\cdot\sum_{v\in H}\#_v(f)\\ &\geq \frac{1}{2}\cdot\sum_{v:\#_v(f)>m/n}\left(\#_v(f)-\frac{m}{n}\right)\end{aligned}$$

which means that f' is $\epsilon/2$-far from the equal-frequencies property. Note that the probability that f is rejected by the verifier is lower-bounded by the probability that the verifier rejects f', since each collision in f' is also a collision in f. Hence, we may just analyze the latter probability while taking advantage of $\#_w(f')<4\cdot m/n$ (rather than $\#_w(f')\leq\frac{2m}{p}\cdot\log_2 n$). Furthermore, with high probability over the choice of the sample $(i_1,...,i_p)$ in Step 1, no value of f' occurs in the sample more than $\log_2 n$ times, which means that Step 3a can be avoided. Hence, we get

Theorem 3.3 (ds-IPP for the equal-frequencies property, Theorem 1.5 restated): *For every $\alpha\in(0,0.5)$, the equal-frequencies property* (of functions from $[m]$ to $[n]$) *has an IPP the query complexity of the verifier is $O(n^{0.5-\alpha})$, the query complexity of the honest prover is $p=\widetilde{O}(n^{0.5+\alpha}/\epsilon^2)$, and the communication complexity is $O(p\cdot\log N)$. Furthermore, if $p=o(n^{1-\Omega(1)})$, then the query complexity of the honest prover can be reduced to $O(n^{0.5+\alpha}/\epsilon^2)$.*

The condition in the furthermore clause holds whenever $\epsilon \geq n^{-(0.5-\alpha)/2}$. To justify the furthermore clause, assume that $p = o(n^{1-(1/t})$ for some $t \in \mathbb{N}$, and consider an execution with input f' (as in the foregoing discussion). Then, with high probability, the sample does not contain a t-way collision of f', and so avoiding rejection in Step 3 requires cheating on $\Delta/(t-1)$ values (rather than on $\Delta/\log_2 n$ values). This allows using $p = O((n/\epsilon)^{0.5+\alpha})$ rather than $p = \widetilde{O}((n/\epsilon)^{0.5+\alpha})$.

3.2 Obtaining IPPs for fixed-frequency properties via reductions

As stated before, our main observation here is that the reductions (presented in [13, Sec. 11.2.2]) from testing equality to any fixed distribution D to testing uniformity apply also in the context of ds-IPPs.

The aforementioned reductions are based on "filters" that randomly map elements of one distribution (over $[n]$) to another distribution (over $[n]$) such that repeated invocations of the filter $F = F^D$ on the same element e yields independent samples of $F(e)$. When reducing testing equality to the distribution $\mathcal{D}$ to testing uniformity, one uses a filter F, which depends (of course) on $\mathcal{D}$, such that if X is distributed as $\mathcal{D}$ then $F(X)$ is uniformly distributed over $[n]$, whereas if X is ϵ-far being distributed according to $\mathcal{D}$ then $F(X)$ is $\Omega(\epsilon)$-far from the uniform distributed over $[n]$.

We shall reduce the construction of ds-IPPs for ρ-frequency property to the construction of ds-IPPs for the equal-frequency property by using the filter $F = F^D$ (that is used to reduced testing $\mathcal{D}$ to testing uniformity), where $\mathcal{D}(v) = \rho(v)$ for every $v \in [n]$. Towards presenting this reduction, suppose that F uses randomness $r \in R$, and let $F_r(x)$ denote its output on input x and coins r. We shall use an integer M that is a multiple of $|R|$ such that $M = \omega(n^2)$.

The reduction consists of selecting a random $\frac{M}{|R|}$-to-1 mapping $\mu : [M] \to R$ and defining $f_\mu : [m] \times [M] \to [n]$ such that $f_\mu(i,j) = F_{\mu(j)}(f(i))$, where f is the function that that is input to the IPP for ρ-frequency (and f_μ will be an emulated input to the IPP for equal-frequency). Thus, for every $\frac{M}{|R|}$-to-1 mapping $\mu : [M] \to R$ and for for every $i \in [m]$ and $w \in [n]$, it holds that $|\{j \in [M] : f_\mu(i,j) = w\}| = \Pr[F(f(i)) = w] \cdot M$. It follows that

$$\frac{\#_w(f_\mu)}{m \cdot M} = \sum_{v \in [n]} \frac{\#_v(f)}{m} \cdot \Pr[F(v) = w]. \tag{2}$$

The IPP for ρ-frequencies will proceed as follows. On input $f : [m] \to [n]$, the parties select a random μ on-the-fly (see details below), and invoke the equal-frequencies IPP on input $f_\mu : [m] \times [M] \to [n]$, and emulate the queries as follows: When an emulated party for the latter IPP makes the query $(i,j) \in [m] \times [M]$ to f_μ, the corresponding party makes the query i to f and answers the emulated party with the value $F_{\mu(j)}(f(i)) = f_\mu(i,j)$. Once the original verifier makes a decision, the constructed verifier decides accordingly.

As stated above, the parties select μ at random on-the-fly; actually, the selection is performed by the verifier, and whenever the prover needs the value of μ at some $j \in [M]$, it just asks the verifier for $\mu(j)$.

We stress that, due to the choice of $M = \omega(n^2)$ and our focus on $o(n)$-query parties, the queries $(i_1, j_1), ..., (i_t, j_t)$ made to f_μ, for a random μ as above, yield a distribution of answers that is very close to the distribution of $F(f(i_1)), ..., F(f(i_t))$. This is the case because $\mu(j_1), ..., \mu(j_t)$ are almost uniformly and independently distributed. This means that the verdict of our system on input f is very close to the verdict of the equal-frequencies system invoked on f_μ for a random $\frac{M}{|R|}$-to-1 mapping $\mu : [M] \to R$.

On the other hand, by Eq. (2), if the frequencies of f are represented by a random variable X, then (for every μ as above) the frequencies of f_μ are represented by the random variable $F(X)$. Using the hypothesis regarding F (i.e., that it maps $\mathcal{D}$ to a uniform distribution over $[n]$ while mapping X that is ϵ-far from $\mathcal{D}$ to a distribution that is $\Omega(\epsilon)$-far from uniform over $[n]$), we get a reduction from ds-IPPs for fixed-frequency properties to a ds-IPP for the equal-frequency property.

Note that this reduction preserves the query complexities of both parties, while increasing the communication complexity in a way that depends on the IPP for equal-frequencies. Specifically, if the communication complexity of the original IPP is logarithmic in the domain of the function, as is the case in the IPP of used in proving Theorem 3.3, then so is the derived IPP. This assertion relies on the fact that the filters presented in [13, Sec. 11.2.2] have logarithmic randomness complexity (and so we can use M such that $\log_2 M = O(\log n)$). Combining the foregoing reduction with Theorem 3.3, we get

Theorem 3.4 (ds-IPP for fixed-frequency properties): *For $\rho : [n] \to [0,1]$ such that $\sum_{v\in[n]} \rho(v) = 1$ and every $\alpha \in (0, 0.5)$, the ρ-frequencies property* (of functions from $[m]$ to $[n]$) *has an IPP the query complexity of the verifier is $O(n^{0.5-\alpha})$, the query complexity of the honest prover is $p = \widetilde{O}(n^{0.5+\alpha}/\epsilon^2)$, and the communication complexity is $O(p \cdot \log m)$. Furthermore, if $p = o(n^{1-\Omega(1)})$, then the query complexity of the honest prover can be reduced to $O(n^{0.5+\alpha}/\epsilon^2)$.*

We comment that the $\frac{M}{|R|}$-to-one mapping μ was selected at random only for the sake of presenting a general reduction. This is not required in the case that the ds-IPP for equal-frequencies is "location oblivious" (as the ds-IPP presented in Section 3.1). Specifically, the distribution of the verifier and (honest) prover queries (coupled with their communication) is invariant when applying any fixed permutation to the queries (and the corresponding communication). Recall that in the ds-IPP presented in Section 3.1, the prover's queries form a random p-subset of the function's domain, which is provided by the verifier, and the verifier's queries are a random q-subset of the former subset.

4 IPPs for the flat-frequencies property: Proof of Theorem 1.7

Recall that $f : [m] \to [n]$ is a flat-frequencies function if f is a ρ-frequencies function for a frequency function ρ such that for some $k \in \mathbb{N}$ and every $v \in [n]$ it holds that $\rho(v) \in \{0, 1/k\}$. In this case we say that f is k-flat. Indeed, the equal-frequencies functions correspond to the special case of flat-frequency in which $k = n$. The set of flat-frequencies functions is called the flat-frequencies property.

For simplicity, let us first assume that k is given (i.e., we focus on testing k-flat functions). In this case, the task is testing whether $f : [m] \to [n]$ has k values such that each value occurs m/k times. This is analogous to testing uniformity over an k-subset of $[n]$, which has sample complexity $\Theta(k^{2/3})$ [5]. Recall that there are no ds-IPPs for uniformity over k-subsets of $[n]$; an IPP for this property of distributions in which the verfier uses $o(k^{2/3})$ samples requires the prover to take $\Omega(k)$ samples [26].

Towards ds-IPPs for flat-frequencies functions. The tester of uniformity over an k-subset [5] first approximates the collision-probability of the distribution and if it matches the uniform case (i.e, is $1/\sqrt{k}$), then it approximate the 3-way collision probability of the distribution. So the point is having the ds-IPP check 3-way-collisions (in samples of the function's value), and it seems that the strategy used for 2-way collisions should suffice.

Actually, the ds-IPP for flat-frequencies offers a better verifier-vs-prover query trade-off than the ds-IPP for equal-frequencies, because the number of 3-way collisions grows cubically with the number of samples. Hence, if the main sample is of size $p = \widetilde{O}(k^{(2/3)+\alpha})$, then the number of 3-way collisions in the sample of an k-flat function is of the form $\Omega(p^3/k^2) = k^{3\alpha}$, and so we only need to subsample at density $k^{-3\alpha}$, which means $\widetilde{O}(k^{(2/3)-2\alpha})$ queries for the verifier. Noting that $\alpha < 1/3$ anyhow, this suffices also for the pairwise collisions, because

$$k^{(2/3)+\alpha} \cdot k^{(2/3)-2\alpha} = k^{1+(1/3)-\alpha} \gg k.$$

The ds-IPPs for k-flat functions. Let $\delta = \mathrm{poly}(\epsilon)$. For every constant $\alpha \in (0, 1/3)$, we present an IPP in which the prover makes $p = \widetilde{O}((k/\delta)^{(2/3)+\alpha})$ queries whereas the verifier makes $q = O((k/\delta)^{(2/3)-2\alpha})$ queries. Again, we assume (w.l.o.g.) that $p < m$.

Construction 4.1 (ds-IPP for flat-frequency properties): *On input m, n, k and ϵ, and oracle acess to $f : [m] \to [n]$, the proof system proceeds as follows.*

1. *The verifier selects $i_1, ..., i_p \in [m]$ uniformly at random and sends $(i_1, ..., i_p)$ to the prover.*
2. *For every $j \in [p]$, the prover obtains $v_j \leftarrow f(i_j)$, and sends $(v_1, ..., v_p)$ to the verifier.*
3. *The verifier subjects the received p-long sequence $(v_1, ..., v_p)$ to three checks:*
   (a) *If some value occurs more than $\log_2 n$ times in the sequence* (i.e., if $|\{j \in [p] : v_j = v\}| > \log_2 n$ for some $v \in [n]$), *then the verifier rejects.*
   (b) *If the empirical collision probability in the p-long sequence is not $(1 \pm \delta)/k$, then the verifier rejects. That is, the verifier rejects if $|\{\{a, b\} \in \binom{[p]}{2} : v_a = v_b\}|$ is not $\frac{1\pm\delta}{k} \cdot \binom{p}{2}$.*
   (c) *If the empirical 3-way collision probability in the p-long sequence exceeds $(1 + \delta)/k^2$, then the verifier rejects. That is, the verifier rejects if $|\{\{a, b, c\} \in \binom{[p]}{3} : v_a = v_b = v_c\}|$ is greater than $\frac{1\pm\delta}{k^2} \cdot \binom{p}{3}$.*

   Otherwise (i.e., in case the verifier did not reject), *the verifier proceeds to the next step.*
4. *The verifier selects uniformly at random a q-subset $J \subset [p]$ and accepts if and only if $v_j = f(i_j)$ for every $j \in J$.*

If f is an k-flat function, then, with high probability, the correct sequence $(f(i_1), ..., f(i_p))$ passes all checks of Step 3. Indeed, in such a case, the collision probability is $1/k$ and the 3-way collision probability is $1/k^2$, whereas (by choice of p) the empirical pairwise and 3-way collision probabilities are within a factor of $1 \pm \delta$ of the actual value (see Appendices A.2 and A.3).[10]

The analysis of the case that f is ϵ-far from the property is based on [5, Lem. 3.4] that asserts the following: *If the pairwise collision probability of a distribution D is $(1 \pm \delta)/k$ and its 3-way collision probability is at most $(1 + \delta)/k^2$, then D is $O(\delta^{1/3})$-close to being uniform on some set.*

[10] When using Appendix A.3, we consider a 3-uniform hypergraph over vertex set $[m]$ such that $\{x, y, z\}$ is a hyper-edge if $f(x) = f(y) = f(z)$. In the current case, there are at least m^3/k^2 hyper-edges and each vertex (resp., pair of vertices) participates in at most $O((m^2/p^2) \log n)$ (resp., $O((m/p^2) \log n)$) hyper-edges.

The original proof actually establishes that this set has size $(1 \pm O(\delta^{1/3}))/k$, and it follows that D is $O(\delta^{1/3})$-close to being uniform on a set of size k. We shall show that if the function f is accepted with high probability, then f is $O(\delta^{1/3})$-close to being k-flat.

We say that a value v as heavy if $\#_v(f) > (3m \cdot \log_2 n)/p$, and let $H \overset{\text{def}}{=} \{v \in [n] : \#_v(f) > (3m \cdot \log_2 n)/p\}$ and $L = [n] \setminus H$. Analogously to the alternative analysis in Section 3.1 (which yields a proof of Theorem 3.3), we decompose $f : [m] \to [n]$ into $f_H : f^{-1}(H) \to [n]$ and $f_L : f^{-1}(L) \to [n]$ such that $f(i) = f_H(i)$ if $i \in f^{-1}(H)$ and $f(i) = f_L(i)$ otherwise. We may assume that $\Pr_{i\in[m]}[f(i) \in H] = O(1/q)$, since otherwise (w.h.p) the verifier rejects either in Step 3a (if the prover answers truthfully on most samples with f-values in H)[11] or in Step 4 (if the prover cheats on most of these values). Hence, we may analyze the execution on f_L rather than on f, because, for $t \in \{2, 3\}$, the difference (due to heavy values) in the claimed number of t-way collisions in the sample is at most $O(p/q) \cdot (\log_2 n)^{t-1} = o(p^t \cdot \delta/k^{t-1})$. (The bound on the number of t-collision follows from the fact that (whp) the sample contains at most $O(p/q)$ heavy values, whereas the inequality uses $p \cdot q = \omega((k/\delta) \log^2 n)$, which implies $O(p/q) \cdot (\log_2 n)^2 = o(p^2 \cdot \delta/k)$ and $O(p/q) \cdot (\log_2 n)^3 = o(p^3 \cdot \delta/k^2)$.)

The analysis of the execution on input f_L proceeds analogously to the simplified analysis in Section 3.1. The key observation is that the empirical pairwise and 3-way collision probabilities under f_L are within a factor of $1 \pm \delta$ of their actual value (see Appendices A.2 and A.3). Hence, the fact that both Step 3b (resp., Step 3c) and Step 4 accept (w.h.p) implies that the pairwise (resp., 3-way) collision probability under f_L is $(1 \pm 3\delta)/k$ (resp., at most $(1 + 3\delta)/k^2$).[12] Applying [5, Lem. 3.4] it follows that f_L (and so also f) is $O(\delta^{1/3})$-close to being m-flat. Thus, we get

Theorem 4.2 (ds-IPP for the flat-frequencies property, Theorem 1.7 restated): *For $k \in [n]$ and every $\alpha \in (0, 1/3)$, the k-flat property* (of functions from $[m]$ to $[n]$) *has an IPP in which the query complexity of the verifier is* $\text{poly}(1/\epsilon) \cdot k^{(2/3)-2\alpha}$*, the query complexity of the prover is* $p = \text{poly}(1/\epsilon) \cdot \widetilde{O}(k^{(2/3)+\alpha})$*, and the communication complexity is* $O(p \cdot \log m)$*.*

Although constructing ds-IPPs for flat-frequency does not reduce to constructing ds-IPPs for k-flatness, we observe that the specific ds-IPP for k-flatness (i.e., of Construction 4.1) can be used in the straightforward reduction. In this reduction, the honest prover first determines a $1 \pm 0.1 \cdot \delta$ factor approximation of the size of the image of the flat-frequencies function, denoted $\widetilde{k}$, sends $\widetilde{k}$ to the verifier, and the two parties proceed with Construction 4.1.[13] The point is that the latter ds-IPP is insensitive to such a small deviation in the size of the image; that is, if f is m-flat, then the verifier of Construction 4.1. accepts (whp) even when the parties use the size parameter $\widetilde{k} = (1 \pm 0.1 \cdot \delta) \cdot k$ (instead of k).

5 IPPs for frequency-bounded functions

In retrospect, constructing IPPs for frequency-bounded functions (which are analogous to "probability bounded distributions") is implicit in Sections 3.1 and 4 (see Step 3a in the ds-IPPs presented there). We detail this IPP here.

[11] Since in such a case, each heavy value is likely to appear more than $2 \log_2 n$ times in the sample.

[12] Specifically, a larger deviation in the t-way collision probability would translate to a deviation of at least $\Omega(p^t \cdot \delta/k^{t-1}) = (p/q) \cdot \Omega(q \cdot \delta \cdot (p/k)^{t-1})$ such collisions in the empirical frequencies (in the p-sized sample. Note that $q \cdot \delta \cdot (p/k) \gg (k/\delta)^{(1/3)-\alpha}$ whereas $q \cdot \delta \cdot (p/k)^2) \gg (1/\delta)^2$.

[13] This approximation is obtained by using the tester of [5], which has sample complexity $\text{poly}(\delta) \cdot \widetilde{k}^{2/3}$.

Definition 5.1 (frequency bounded functions): *For $\tau \in (0,1]$, we say that $f : [m] \to [n]$ is* τ-bounded *if for every $v \in [n]$ it holds that $\#_v(f) \leq \tau \cdot m$. where $\#_v(f) \stackrel{\text{def}}{=} |\{i \in [m] : f(i) = v\}|$.*

We mention that testing τ-bounded functions requires at least $(1/\tau)^{1-\Omega(1)}$ queries. This follows from the work of [21]; see Appendix A.4 for details. In contrast, we present a doubly-sublinear IPP for τ-bounded functions in which the prover makes $p = \widetilde{O}(1/\tau)$ queries and the verifier makes $O(1/\epsilon)$ queries.

Construction 5.2 (ds-IPP for frequency-bounded functions): *On input m, n, τ and ϵ, and oracle access to $f : [m] \to [n]$, the proof system proceeds as follows.*

1. *The verifier selects $i_1, ..., i_p \in [m]$ uniformly at random and sends $(i_1, ..., i_p)$ to the prover.*
2. *For every $j \in [p]$, the prover obtains $v_j \leftarrow f(i_j)$, and sends $(v_1, ..., v_p)$ to the verifer.*
3. *If some value occurs more than $(1 + 0.1 \cdot \epsilon) \cdot \tau \cdot p$ times in the p-long sequence received by the verifier, then the verifier rejects. Otherwise, the verifier selects uniformly at random a $O(1/\epsilon)$-subset $J \subset [p]$ and accepts if and only if $v_j = f(i_j)$ for every $j \in J$.*

If f is τ-bounded, then, whp, the sequence selected by the verifier is $(1 + 0.1 \cdot \epsilon) \cdot \tau$-bounded (i.e., $|\{j \in [p] : f(i_j) = v\}| \leq (1 + 0.1\epsilon) \cdot \tau \cdot p$ fo every $v \in [n]$). In this case, the prover just provides the true values and the verifier accepts. On the other hand, if f is ϵ-far from τ-bounded, then

$$\sum_{v : \#_v(f) > \tau \cdot m} (\#_v(f) - \tau \cdot m) > \epsilon \cdot m.$$

In this case, with high probability over the choice of $i_1, ..., i_p \in [m]$, for $g : [p] \to [n]$ such that $g(j) = f(i_j)$ it holds that

$$\sum_{v : \#_v(g) > \tau \cdot p} (\#_v(g) - \tau \cdot p) > 0.5 \cdot \epsilon \cdot p.$$

Hence, to avoid upfront rejection (per the initial counting performed by the verifier), the prover must provide wrong answers on more than $0.4 \cdot \epsilon \cdot p$ of the indices. But, in that case, the verifier rejects w.h.p. Hence, we get

Theorem 5.3 (IPP for frequency bounded functions): *For every $\tau \in (0,1]$, there exist an IPP for τ-bounded functions in which the prover makes $p = \widetilde{O}(1/\tau^{-1})$ queries and the verifier makes $O(1/\epsilon)$ queries. When the functions are over the domain $[m]$, the communication complexity is $O(p \cdot \log m)$.*

The reason that this result is relevant to the study of ds-IPPs is that when constructing a ds-IPP system we may set τ such that $\widetilde{O}(1/\tau^{-1})$ does not exceed the query complexity of the intended prover. Indeed, this was done implicitly in Section 4, where we implicitly used Theorem 5.3 with τ that inversely proportional to the query complexity of the intended prover and a proximity parameter set to $1/q$ (see Step 3a in Construction 4.1), Executing this τ-bounded IPP yields complexities that are affordable per Theorem 4.2.

6 IPPs for all locally-invariant properties: Proof of Theorem 1.8

As outlined in Section 1.4, our basic strategy is to have the verifier learn the frequency function of the input function $f : [m] \to [n]$ (i.e., the function $\rho : [n] \to [0,1]$ s.t. $\rho(v) = \#_v(f)/m$ for each v) by delegating the corresponding $O(n/\epsilon^2)$ queries to the prover. Indeed, we use the fact that any distribution D over $[n]$ can be learned using $O(n/\epsilon^2)$ samples of D (see [13, Exer. 11.4]).

Construction 6.1 (IPP for the locally-invariant property $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$): *On input m, n and ϵ, and oracle access to $f : [m] \to [n]$, the proof system proceeds as follows.*

1. *For $p = O(n/\epsilon^2)$, the verifier selects $i_1, ..., i_p \in [m]$ uniformly at random, and sends $(i_1, ..., i_p)$ to the prover.*
2. *For every $j \in [p]$, the prover obtains $v_j \leftarrow f(i_j)$, by querying f, and sends $(v_1, ..., v_p)$ to the verifier.*
3. *The verifier computes $\rho : [n] \to [0,1]$ such that $\rho(v) = |\{j \in [p] : v_j = v\}|/p$ for every $v \in [n]$. If ρ is $\epsilon/2$-far from the frequency function of each function in $\Pi_{m,n}$, then the verifier rejects; that is, the verifier rejects if for every $g \in \Pi_{m,n}$ it holds that $\sum_v |\rho(v) - \#_v(g)| > \epsilon$. Otherwise* (i.e., in case the verifier did not reject)*, the verifier proceeds to the next step.*
4. *For $q = O(1/\epsilon)$, the verifier selects uniformly at random a q-subset $J \subset [p]$, queries f on each element of $\{i_j : j \in J\}$, and accepts if and only if $v_j = f(i_j)$ for every $j \in J$.*

We first observe that, with high probability, the function ρ computed in Step 3 satisfies $\sum_v |\rho(v) - \#_v(f)| < \epsilon/2$. Hence, the verifier accepts each $f \in \Pi_{m,n}$ with high probability. Turning to the case that $f : [m] \to [n]$ is ϵ-far from $\Pi_{m,n}$, with high probability (over the coice of $i_1, ..., i_p \in [m]$), it holds that

$$\sum_v \left| \frac{\#_v(f(i_1), ..., f(i_p))}{p} - \#_v(f) \right| < \epsilon/2,$$

where $\#_v(v_1, ..., v_p) = |\{j \in [p] : v_j = v\}|$ for every $v \in [n]$. Hence, to avoid rejection in Step 3, the prover must provide wrong values on at least $\epsilon \cdot p/2$ of the p indices. But such cheating will be detected, with high probability, by the verifier (in Step 4).

Appendices

In Appendix A.1 we recall the standard definitions of property testing and distribution testing. Appendices A.2 and A.3 provide proofs of concentration bounds that are used in the main text. In contrast, Appendix A.4 merely justifies the value of an IPP for frequency-bounded functions.

A.1 Property testing and distribution testing

Property testing (or testing properties of functions) studies probabilistic algorithms of sub-linear complexity for deciding whether a given function has a predetermined property or is far from any function having this property. These functions may represent bit-strings (or sequences over larger alphabet), graphs, collections of points (in metric spaces), etc. (The representation of such objects by functions may be redundant.) The aforementioned algorithms, called testers, obtain local views of the function by *performing queries*, and the primary complexity measure is the *query complexity*. The query complexity is typically stated in terms of the size of the function (denoted m below) and a proximity parameter (denoted ϵ), which refers to a distance measure that (combined with the proximity parameter) determines which functions are considered far from the property. The vanilla definition, which corresponds to the (relative) Hamming distance, is reproduced next.

Definition A.1 (a tester for property Π):[14] *Let $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ such that $\Pi_{m,n}$ contains functions of the form $f : [m]\to[n]$. A* tester for Π *is a probabilistic oracle machine, denoted T, that, on input parameters m, n and ϵ and oracle access to a function $f : [m]\to[n]$, outputs a binary verdict that satisfies the following two conditions.*

1. T accepts inputs in Π: *For every $m, n \in \mathbb{N}$ and $\epsilon > 0$, and for every $f \in \Pi_{m,n}$, it holds that*

$$\Pr[T^f(m,n,\epsilon)=1] \geq 2/3.$$

If $\Pr[T^f(m,n,\epsilon)=1] = 1$ always holds, then we say that T has one-sided error*; otherwise, we say that T has* two-sided error*.*

2. T rejects inputs that are ϵ-far from Π: *For every $m, n \in \mathbb{N}$ and $\epsilon > 0$, and for every $f : [m]\to[n]$ such that $\delta_\Pi(f) > \epsilon$, it holds that*

$$\Pr[T^f(m,n,\epsilon)=0] \geq 2/3,$$

where $\delta(f,g) \overset{\text{def}}{=} |\{i\in[n] : f(i)\neq g(i)\}|/n$ and $\delta_\Pi(f) \overset{\text{def}}{=} \min_{g\in\Pi_n} \{\delta(f,g)\}$.

The query complexity *of T is a function* (of m, n and ϵ) *that specifies the number of queries made by T on input parameters m, n and ϵ, when given oracle access to any function $f : [m]\to[n]$.*

We stress that, while the foregoing definition underlies most studies of property testing, cases in which the distance measure is not uniform over the domain are not rare.[15]

Distribution testing (or testing properties of distributions) studies probabilistic algorithms of sub-linear complexity for deciding whether a given distribution has a predetermined property or is far from any distribution having this property. The corresponding testers obtain samples of the tested distribution, which we view as local views of the distribution, and the primary complexity measure is the *sample complexity*, which is stated in terms of the domain of the distribution and a proximity parameter (denoted ϵ), which determines which distributions are considered far from the property. In this case, the distance measure is the total variation distance between distributions; that is, for distributions X and Y, we let $\delta(X,Y) \overset{\text{def}}{=} \frac{1}{2}\cdot\sum_z |\Pr[X=z] - \Pr[Y=z]|$, which equals $\max_S\{\Pr[X\in S] - \Pr[Y\in S]\}$. We say that X is ϵ-far from Y if $\delta(X,Y) > \epsilon$.

Definition A.2 (testing properties of distributions): *Let $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ be a property of distributions such that $\mathcal{D}_n$ is a set of distributions over $[n]$, and $s : \mathbb{N}\times(0,1]\to\mathbb{N}$. A* tester*, denoted T, of* sample complexity s for the property $\mathcal{D}$ *is a probabilistic machine that, on input parameters n and ϵ, and a sequence of $s(n,\epsilon)$ samples drawn from an unknown distribution X over $[n]$, satisfies the following two conditions.*

[14] Since our focus is on properties of the form $\Pi = \bigcup_{m,n\in\mathbb{N}} \Pi_{m,n}$ (as in Definition 1.1), we provide all machines with two size parameters (i.e., m and n). In addition, the complexity is stated in terms of m, n and ϵ.

[15] Examples include weighted Hamming distance, edit distance, and $\mathcal{L}_1$-distance (see, e.g., [13, Sec. 12.4]).

1. The tester accepts distributions that belong to $\mathcal{D}$: *If X is in $\mathcal{D}_n$, then*

$$\mathrm{Pr}_{x_1,...,x_s \sim X}[T(n,\epsilon;x_1,...,x_s)\!=\!1] \geq 2/3,$$

where $s = s(n,\epsilon)$ and $x_1,...,x_s$ are drawn independently from the distribution X.

2. The tester rejects distributions that are far from $\mathcal{D}$: *If X is ϵ-far from any distribution in $\mathcal{D}_n$* (i.e., X is ϵ-far from $\mathcal{D}$), *then*

$$\mathrm{Pr}_{x_1,...,x_s \sim X}[T(n,\epsilon;x_1,...,x_s)\!=\!0] \geq 2/3,$$

where $s = s(n,\epsilon)$ and $x_1,...,x_s$ are as in the previous item.

An alternative formulation provides the tester with oracle access to a sampling device rather than to $s(n,\epsilon)$ samples. This alternative is streamlined with Definition A.1.

A.2 Probabilistic analysis of pairwise collisions

A technical issue that arises in the proofs of Theorems 2.1, 3.3 and 4.2 is that these proofs rely on concentration bounds regarding a set of events, where each event refers to a pair of samples; specifically, the events are collisions of the image of the two samples under a tested function. The concentration bound follows by the fact that the relevant events are sufficiently independent. Specifically, as stated in Footnote 7, most pairs of events refer to four independent samples, whereas the few pairs of events that refer to three independent samples correspond to three-way collisions and occur with very small probability.

For sake of good order, we provide a detailed analysis of this situation. Actually, to make the analysis more useful for other applications, we abstract the situation by viewing the relevant events as edges of a graph. We claim that when the maximal degree of vertices in the graph is not much larger than the average degree, the edge density in a sufficiently large induced subgraph approximates the edge density of the graph. In our applications, the vertex-set represents the domain of the function $f : [m] \to [n]$ and the edges of the graph correspond to collisions under f. Hence, in our application, the graph consists of a collection of isolated cliques (and the number of samples is p). (In the proof of Theorem 2.1 the maximum degree of the graph is 1 (whereas the average degree is at least $\epsilon \gg 1/p$); in the proof of Theorems 3.3 and 4.2 the maximum degree is $O(\frac{m}{p} \cdot \log n)$ (whereas the average degree is at least m/n and m/k, resp.).)

Claim:[16] *Let $G = (V,E)$ be a graph of maximun degree d, and let $\rho = 2|E|/|V|^2$. Then, for sufficiently large $s = O(\max(\frac{\rho^{-1/2}}{\eta}, \frac{d|V|}{\eta^2|E|}))$, selecting a multi-set $U = \{u_1,...,u_s\}$ of s vertices uniformly at random, with very high probability, the subgraph of G induced by U has $(1 \pm \eta)\cdot\rho\cdot\binom{m}{2}$ edges; that is,*

$$\mathrm{Pr}_{u_1,...,u_s \in V}\left[\left|\left\{\{i,j\}\!\in\!\binom{[s]}{2} : \{u_i,u_j\}\!\in\! E\right\}\right| \neq (1\pm\eta)\cdot\rho\cdot\binom{s}{2}\right] \leq \frac{O(1)}{\eta^2\cdot\rho\cdot s^2} + \frac{O(d)}{\eta^2\cdot(|E|/|V|)\cdot s}$$

Note that the first term represents the upper bound that would have applied if the vertex-pairs were pairwise independent. The second term is an error term and its size depends on the ratio between the maximal and average degree of vertices in G (i.e., the ratio of d over $2|E|/|V|$).

[16] An analogous claim can be proved for the case of sampling without repetitions (i.e., a random set $U \in \binom{V}{s}$ rather than a random multi-set $U \in V^s$.

Proof: For $i, j \in [s]$, let $\zeta_{i,j}$ be a random variable such that $\zeta_{i,j} = 1$ if $\{u_i, u_j\} \in E$ and $\zeta_{i,j} = 0$ otherwise. Then, $\text{Exp}[\zeta_{i,j}] = \rho$. Letting $P = \binom{[s]}{2}$, and applying Chebyshev's inequality, we get

$$\begin{aligned}
\Pr\left[\left|\sum_{\{i,j\}\in P} \zeta_{i,j} \; - \rho \cdot |P|\right| \geq \eta \cdot \rho \cdot |P|\right] \;&\leq\; \frac{\text{Var}\left[\sum_{\{i,j\}\in P} \zeta_{i,j}\right]}{(\eta \cdot \rho \cdot |P|)^2} \\
&\leq \sum_{\{i,j\}\in P} \frac{\text{Var}[\zeta_{i,j}]}{\eta^2 \cdot (\rho \cdot |P|)^2} \;+\; \sum_{\{i_1,j_1\}\neq\{i_2,j_2\}\in P} \frac{\text{Var}[\zeta_{i_1,j_1}\zeta_{i_2,j_2}]}{\eta^2 \cdot (\rho \cdot |P|)^2} \qquad (3)
\end{aligned}$$

Using $\text{Var}[\zeta_{i,j}] \leq \text{Exp}[\zeta_{i,j}^2] = \text{Exp}[\zeta_{i,j}] = \rho$, the first sum is upper-bounded by $1/(\eta^2 \cdot \rho \cdot |P|)$. Letting $\overline{\zeta}_{i,j} = \zeta_{i,j} - \text{Exp}[\zeta_{i,j}]$ and using $\text{Var}[\zeta_{i_1,j_1}\zeta_{i_2,j_2}] = \text{Exp}[\overline{\zeta}_{i_1,j_1}\overline{\zeta}_{i_2,j_2}]$, observe that $\text{Var}[\zeta_{i_1,j_1}\zeta_{i_2,j_2}]$ equals 0 if $|\{i_1, j_1, i_2, j_2\}| = 4$ and is upper-bounded by $\text{Exp}[\zeta_{i_1,j_1}\zeta_{i_2,j_2}]$ otherwise (i.e., when $|\{i_1, j_1, i_2, j_2\}| = 3$). In the latter case, we use $\text{Exp}[\zeta_{i_1,j_1}\zeta_{i_2,j_2}] = \text{Exp}[\zeta_{i_1,j_1}] \cdot \Pr[\zeta_{i_2,j_2}=1|\zeta_{i_1,j_1}=1]$. Hence, Eq. (3) is upper-bounded by

$$\frac{1}{\eta^2 \cdot \rho \cdot |P|} + \frac{O(|P| \cdot s)}{\eta^2 \cdot (\rho \cdot |P|)^2} \cdot \rho \cdot \Pr[\zeta_{2,3}=1|\zeta_{1,2}=1]. \qquad (4)$$

Using the fact that the maximal degree in G is d, it follows that $\Pr[\zeta_{2,3}=1|\zeta_{1,2}=1] \leq d/|V|$, since fixing u_1, u_2 such that $\{u_1, u_2\} \in E$, the probability that a random $u_3 \in V$ is a neighbor of u_2 is at most $d/|V|$. Hence, Eq. (4) is upper-bounded by

$$\frac{1}{\eta^2 \cdot \rho \cdot |P|} + \frac{O(s)}{\eta^2 \cdot \rho \cdot |P|} \cdot \frac{d}{|V|} \;=\; \frac{1}{\eta^2 \cdot \rho \cdot |P|} + \frac{O(1)}{\eta^2 \cdot s} \cdot \frac{d}{|E|/|V|}$$

since $\rho = 2|E|/|V|^2$ (and $|P| = \Omega(s^2)$). The claim follows. ■

Our applications. The foregoing claim is used in Sections 2, 3.1 and 4. In all cases, the vertex-set V is $[m]$ and the edge-set E corresponds to collisions under the tested function that ranges over $[n]$. Hence, the average degree $2|E|/|V|$ is $\Omega(m/n)$, whereas the maximum degree d is $O((m/p)\log n)$. Recalling that $s = p$, for Sections 2 and 3.1, the second (i.e., error) term in the probability bound of the claim is

$$\frac{O(d)}{\eta^2 \cdot (|E|/|V|) \cdot p} \;=\; \frac{O((m/p) \cdot \log n)}{\eta^2 \cdot (m/n) \cdot p} \;=\; \frac{O(n \cdot \log n)}{\eta^2 \cdot p^2}$$

which yields a total probability bound of

$$\frac{O(1)}{\eta^2 \cdot \rho \cdot p^2} + \frac{O(n \cdot \log n)}{\eta^2 \cdot p^2}$$

which is $o(1)$ provided that $p = \omega(\max((n \log n)^{1/2}, \rho^{-1/2})/\eta)$. In Section 4, n is replaced by k. Recall that in Section 2 we use $\rho = \epsilon$ and $\eta = 1/2$, whereas in Section 3.1 (resp., Section 4) we use $\rho = 1/n$ and $\eta = \epsilon^2$ (resp., $\rho = 1/k$ and $\eta = \delta$). Hence, using $p = \widetilde{O}(n^{1/2}/\epsilon^{-2})$ in Sections 2 and 3.1 (resp., $p = \widetilde{O}(k^{1/2}/\epsilon^{-2})$ in Section 4) suffices.

A.3 Probabilistic analysis of 3-way collisions

Analogously to Appendix A.2, we prove the following claim that offer a way to analyze a sequence of $\binom{s}{3}$ events in which each event refers to three samples (in a sequence of s samples). The point is that most pairs of events refer to six independent samples, whereas the few pairs of events that refer to five or less independent samples correspond to events that occur with very small probability.[17]

Claim: *Let $G=(V,E)$ be a 3-uniform hypergraph and $\rho=6|E|/|V|^3$. Suppose that each vertex participates in at most d_1 hyper-edges, and that each pair of vertices participates in at most d_2 hyper-edges.*[18] *Then, for sufficiently large $s=O(\max(\frac{\rho^{-1/3}}{\eta^{2/3}},\frac{d_1}{\eta^2\cdot|E|/|V|},\sqrt{\frac{d_2}{\eta^2\cdot|E|/|V|^2}}))$, selecting a multi-set $U=\{u_1,...,u_s\}$ of s vertices uniformly at random, with very high probability, the subgraph of G induced by U has $(1\pm\eta)\cdot\rho\cdot\binom{s}{3}$ edges; that is,*

$$\begin{aligned}\mathrm{Pr}_{u_1,...,u_s\in V}\left[\left|\left\{\{i,j,k\}\in\binom{[s]}{3}:\{u_i,u_j,u_k\}\in E\right\}\right|\neq(1\pm\eta)\cdot\rho\cdot\binom{s}{3}\right]&\\ \le\ \frac{O(1)}{\eta^2\cdot\rho\cdot s^3}+\frac{O(d_1)}{\eta^2\cdot(|E|/|V|)\cdot s}+\frac{O(d_2)}{\eta^2\cdot(|E|/|V|^2)\cdot s^2}&\end{aligned}$$

The claim can be extended to t-uniform hypergraphs for any constant $t\ge 2$.

Proof: For $i,j,k\in[s]$, let $\zeta_{i,j,k}$ be a random variable such that $\zeta_{i,j,k}=1$ if $\{u_i,u_j,u_k\}\in E$ and $\zeta_{i,j,k}=0$ otherwise. Then, $\mathrm{Exp}[\zeta_{i,j,k}]=\rho$. Letting $T=\binom{[s]}{3}$, and applying Chebyshev's inequality, we get

$$\begin{aligned}\mathrm{Pr}\left[\left|\sum_{\{i,j,k\}\in T}\zeta_{i,j,k}\ -\rho\cdot|T|\right|\ge\eta\cdot\rho\cdot|T|\right]\ &\le\ \frac{\mathrm{Var}\left[\sum_{\{i,j,k\}\in T}\zeta_{i,j,k}\right]}{(\eta\cdot\rho\cdot|T|)^2}\\ &\le\ \sum_{\{i,j,k\}\in T}\frac{\mathrm{Var}[\zeta_{i,j,k}]}{\eta^2\cdot(\rho\cdot|T|)^2}\ +\sum_{\{i_1,j_1,k_1\}\neq\{i_2,j_2,k_2\}\in T}\frac{\mathrm{Var}[\zeta_{i_1,j_1,k_1}\zeta_{i_2,j_2,k_2}]}{\eta^2\cdot(\rho\cdot|T|)^2}\end{aligned}\tag{5}$$

Using $\mathrm{Var}[\zeta_{i,j,k}]\le\rho$, the first sum is upper-bounded by $1/(\eta^2\cdot\rho\cdot|T|)$. Letting $\overline{\zeta}_{i,j,k}=\zeta_{i,j,k}-\mathrm{Exp}[\zeta_{i,j,k}]$ and using $\mathrm{Var}[\zeta_{i_1,j_1,k_1}\zeta_{i_2,j_2,k_2}]=\mathrm{Exp}[\overline{\zeta}_{i_1,j_1,k_1}\overline{\zeta}_{i_2,j_2,k_2}]$, observe that $\mathrm{Var}[\zeta_{i_1,j_1,k_1}\zeta_{i_2,j_2,k_2}]$ equals 0 if $|\{i_1,j_1,k_1,i_2,j_2,k_2\}|=6$ and is upper-bounded by $\mathrm{Exp}[\zeta_{i_1,j_1,k_1}]\cdot\mathrm{Pr}[\zeta_{i_2,j_2,k_2}=1|\zeta_{i_1,j_1,k_1}=1]$ otherwise (i.e., when $|\{i_1,j_1,k_1,i_2,j_2,k_2\}|\in\{4,5\}$). Hence, Eq. (5) is upper-bounded by

$$\frac{1}{\eta^2\cdot\rho\cdot|T|}+\frac{O(|T|\cdot s)}{\eta^2\cdot(\rho\cdot|T|)^2}\cdot\rho\cdot\mathrm{Pr}[\zeta_{2,3,4}=1|\zeta_{1,2,3}=1]+\frac{O(|T|\cdot s^2)}{\eta^2\cdot(\rho\cdot|T|)^2}\cdot\rho\cdot\mathrm{Pr}[\zeta_{3,4,5}=1|\zeta_{1,2,3}=1].\tag{6}$$

Using the hypothesis regarding the maximal "degrees" of vertices and pairs of vertices (i.e., the bounds d_1 and d_2), it follows that $\mathrm{Pr}[\zeta_{2,3,4}=1|\zeta_{1,2,3}=1]\le d_2/|V|$ and $\mathrm{Pr}[\zeta_{3,4,5}=1|\zeta_{1,2,3}=1]\le 2d_1/|V|^2$. Hence, Eq. (6) is upper-bounded by

$$\begin{aligned}&\frac{1}{\eta^2\cdot\rho\cdot|T|}+\frac{O(s)}{\eta^2\cdot\rho\cdot|T|}\cdot\frac{d_2}{|V|}+\frac{O(s^2)}{\eta^2\cdot\rho\cdot|T|}\cdot\frac{2d_1}{|V|^2}\\ &=\ \frac{O(1)}{\eta^2\cdot\rho\cdot s^3}+\frac{O(1)}{\eta^2\cdot s^2}\cdot\frac{d_2}{|E|/|V|^2}+\frac{O(1)}{\eta^2\cdot s}\cdot\frac{d_1}{|E|/|V|}\end{aligned}$$

[17] Again, we refer to sampling with repetitions (i.e., a random multi-set $U\in V^s$), but an analogous claim holds for sampling without repetitions (i.e., a random set $U\in\binom{V}{s}$).

[18] Needless to say, $d_1\ge 2|E|/|V|$ and $d_2\ge 2|E|/\binom{|V|}{2}$.

since $\rho = 3|E|/|V|^3$ and $|T| = \Omega(s^3)$. The claim follows. ■

Our application. The foregoing claim is used in Section 4. In that case, the vertex-set V is $[m]$ and the edge-set E corresponds to 3-way collisions under the tested function that is supposed to be uniform on a k-subset of $[n]$. Hence, $|E| = \Omega(m^3/k^2)$, whereas $d_1 = O((m^2/p^2) \cdot \log n)$ and $d_2 = O((m/p^2) \cdot \log n)$. Recalling that $s = p$, the second term in the probability bound of the claim is

$$\frac{O(d_1)}{\eta^2 \cdot (|E|/|V|) \cdot p} = \frac{O((m^2/p^2) \cdot \log n)}{\eta^2 \cdot (m^2/k^2) \cdot p} = \frac{O(k^2 \cdot \log n)}{\eta^2 \cdot p^3}$$

and the third term is

$$\frac{O(d_2)}{\eta^2 \cdot (|E|/|V|^2) \cdot p^2} = \frac{O((m/p^2) \cdot \log n)}{\eta^2 \cdot (m/k^2) \cdot p} = \frac{O(k^2 \cdot \log n)}{\eta^2 \cdot p^3}$$

which yields a total probability bound of

$$\frac{O(1)}{\eta^2 \cdot \rho \cdot p^3} + \frac{O(k^2 \cdot \log n)}{\eta^2 \cdot p^3}$$

which is $o(1)$ provided that $p = \omega(\max((k \log n)^{2/3}, \rho^{-1/3})/\eta)$. Recalling that $\rho = 1/k^2$ and $\eta = \delta = \text{poly}(\epsilon)$, we may use $p = \text{poly}(1/\epsilon) \cdot k^{2/3}$.

A.4 On testing frequency bounded functions

For every $\beta \in (0.5, 1)$, Goldreich and Ron presented an k^β lower bound on the sample complexity of testing "k-grained" distributions [21]. We first observe that their proof technique allows to prove a similar lower bound on testing *τ-bounded distributions*, which are distributions in which every element occurs with probability at most τ (where $\tau = 1/k$). The latter claim holds because [21] shows that, for every constant $t \in \mathbb{N}$, the following two distributions are indistinguishable by label-invariant algorithms that take $k^{1-(1/2t)}$ samples:

1. A distribution P over $[n]$ such that $P(v) \in \left\{\frac{2j}{2k} : j \in [t+1]\right\} \cup \{0\}$ for every $v \in [n]$.
2. A distribution Q over $[n]$ such that $Q(v) \in \left\{\frac{2j-1}{2k} : j \in [t+1]\right\} \cup \{0\}$ for every $v \in [n]$.

Furthermore, for every $j \in [t+1]$, it holds that $\sum_{v:P(v)=2j/2k} P(j)$ (resp., $\sum_{v:Q(v)=(2j-1)/2k} P(j)$) is either $\Omega(1)$ or 0, where the constant (in the Ω-notation) is independent of k (but does depend on t).

Picking the maximal $i \in \{2, ..., 2t+2\}$ such that $\{v \in [n] : P(v) + Q(v) = i/k\} \neq \emptyset$ (equiv., either $\{v \in [n] : P(v) = i/k\} \neq \emptyset$ or $\{v \in [n] : Q(v) = i/k\} \neq \emptyset$), it follows that one of the two distributions is $\frac{i-1}{k}$-bounded whereas the other distribution is $\Omega(1)$-far from being $\frac{i-1}{k}$-bounded. Letting $\mathcal{P}$ (resp., $\mathcal{Q}$) denote the set of all distributions obtained from P (resp., Q) by permuting the labels, it follows that a ny algorithm that takes $k^{(2t-1)/2t}$ samples fails to distinguish between $\mathcal{P}$ and $\mathcal{Q}$ (cf. [13, Thm. 11.12]). Hence, the sample complexity of testing $O(1/k)$-bounded distributions is at least $k^{(2t-1)/2t}$.

Lastly, using the results of [20, Sec. 6], it follows that the same lower bound applies to the query complexity of testing $O(1/k)$-bounded functions.

References

[1] William Aiello and Johan Hastad. Statistical Zero- Knowledge Languages Can Be Recognized in Two Rounds *Journal of Computer and System Science*, Vol. 42, pages 327–345, 1991.

[2] Noga Alon and Eric Blais. Testing Boolean Function Isomorphism. In *14th RANDOM*, Lecture Notes in Computer Science (Vol. 6302), pages 394–405, 2010.

[3] Noga Alon, Eric Blais, Sourav Chakraborty, David Garcia-Soriano, and Arie Matsliah. Nearly Tight Bounds for Testing Function Isomorphism. *SIAM Journal on Computing*, Vol. 42 (2), pages 459–493, 2013.

[4] Noga Amit, Oded Goldreich and Guy N. Rothblum. Doubly Sub-linear Interactive Proofs of Proximity. In *16th ITCS*, LIPIcs, Volume 325, pages 6:1–6:25, 2025.

[5] Tugkan Batu and Clement L. Canonne. Generalized Uniformity Testing. In *58th FOCS*, pages 880–889, 2017.

[6] Tugkan Batu, Lance Fortnow, Eldar Fischer, Ravi Kumar, Ronitt Rubinfeld, and Patrick White. Testing random variables for independence and identity. In *42nd IEEE Symposium on Foundations of Computer Science*, pages 442–451, 2001.

[7] Tugkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. Testing that distributions are close. *Journal of the ACM*, Vol. 60 (1), pages 4:1–4:25, 2013. Preliminary version in *41st FOCS*, pages 259–269, 2000.

[8] Manuel Blum, Michael Luby and Ronitt Rubinfeld. Self-Testing/Correcting with Applications to Numerical Problems. *Journal of Computer and System Science*, Vol. 47, No. 3, pages 549–595, 1993. Extended abstract *22nd STOC*, 1990.

[9] Alessandro Chiesa and Tom Gur. Proofs of Proximity for Distribution Testing. In *9th ITCS*, LIPIcs, Volume 94, pages 53:1–53:14, 2018.

[10] Ilias Diakonikolas, Daniel Kane, and Alistair Stewart. Sharp Bounds for Generalized Uniformity Testing. In *ECCC*, TR17-132, 2017.

[11] Lance Fortnow. The complexity of perfect zero-knowledge. In *19th STOC*, pages 204–209, 1987.

[12] Peter Gemmell, Richard J. Lipton, Ronitt Rubinfeld, Madhu Sudan, and Avi Wigderson. Self-Testing/Correcting for Polynomials and for Approximate Functions . In *23rd ACM Symposium on the Theory of Computing*, pages 32–42, 1991.

[13] Oded Goldreich. *Introduction to Property Testing.* Cambridge University Press, 2017.

[14] Oded Goldreich. Testing Isomorphism in the Bounded-Degree Graph Model. In *ECCC*, TR19-102, 2019.

[15] Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, pages 653–750, July 1998. Extended abstract in *37th FOCS*, 1996.

[16] Oded Goldreich, Tom Gur, and Ron D. Rothblum. Proofs of proximity for context-free languages and read-once branching programs. *Inf. Comput.*, Vol. 261, pages 175–201, 2018.

[17] Oded Goldreich and Maya Leshkowitz. On Emulating Interactive Proofs with Public Coins. In *ECCC*, TR16-066, 2016.

[18] Oded Goldreich and Dana Ron. Property Testing in Bounded Degree Graphs. *Algorithmica*, Vol. 32 (2), pages 302–343, 2002.

[19] Oded Goldreich and Dana Ron. On Proximity Oblivious Testing. *SIAM Journal on Computing*, Vol. 40, No. 2, pages 534–566, 2011.

[20] Oded Goldreich and Dana Ron. On Sample-Based Testers. *TOCT*, Vol. 8 (2), 2016.

[21] Oded Goldreich and Dana Ron. A Lower Bound on the Complexity of Testing Grained Distributions. *Comput. Complex.*, Vol. 32 (2), Art. 11, 2023.

[22] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating Computation: Interactive Proofs for Muggles. *Journal of the ACM*, Vol. 62 (4), pages 27:1–27:64, 2015. Preliminary version in *40th STOC*, 2008.

[23] Shafi Goldwasser, Silvio Micali and Charles Rackoff. The Knowledge Complexity of Interactive Proof Systems. *SIAM Journal on Computing*, Vol. 18, pages 186–208, 1989. Preliminary version in *17th STOC*, 1985.

[24] Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. Extended abstract in *18th STOC*, 1986.

[25] Tom Gur, Yang P. Liu, Ron D. Rothblum. An Exponential Separation Between MA and AM Proofs of Proximity. *Comput. Complex.*, Vol. 30 (2), Art. 12, 2021.

[26] Tal Herman and Guy N. Rothblum. Lower Bounds on ds-IPPs for distribution testing. In preparation.

[27] Guy N. Rothblum, Salil Vadhan, and Avi Wigderson. Interactive Proofs of Proximity: Delegating Computation in Sublinear Time. In *45th ACM Symposium on the Theory of Computing*, pages 793–802, 2013.

[28] Ronitt Rubinfeld and Madhu Sudan. Self-Testing Polynomial Functions Efficiently and Over Rational Domains. In the proceedings of *3rd SODA*, pages 23–32, 1992.

[29] Ronitt Rubinfeld and Madhu Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal on Computing*, Vol. 25(2), pages 252–271, 1996. Unifies and extends part of the results contained in [12] and [28].

[30] Hadar Strauss. Emulating Computationally Sound Public-Coin IPPs in the Pre-Coordinated Model. In *ECCC*, TR24-131, 2024.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il