Nearly Tight Bounds for Testing Function Isomorphism

Sourav Chakraborty* David García-Soriano† Arie Matsliah‡

Abstract

We study the problem of testing isomorphism (equivalence up to relabelling of the variables) of two Boolean functions $f, g : \{0,1\}^n \to \{0,1\}$. Our main focus is on the most studied case, where one of the functions is given (explicitly) and the other function may be queried.

We prove that for every $k \le n$, the worst-case query complexity of testing isomorphism to a given k-junta is $\Omega(k)$ and $O(k \log k)$. Consequently, the query complexity of testing function isomorphism is $\widetilde{\Theta}(n)$.

Prior to this work, only lower bounds of $\Omega(\log k)$ queries were known, for limited ranges of k, proved by Fischer et al. (FOCS 2002), Blais and O'Donnell (CCC 2010), and recently by Alon and Blais (RANDOM 2010). The nearly tight $O(k \log k)$ upper bound improves on the $\widetilde{O}(k^4)$ upper bound from Fischer et al. (FOCS 2002).

Extending the lower bound proof, we also show polynomial query-complexity lower bounds for the problems of testing whether a function can be computed by a circuit of size $\le s$, and testing whether the Fourier degree of a function is $\le d$. This answers questions posed by Diakonikolas et al. (FOCS 2007).

We also address two closely related problems –

1. Testing isomorphism to a k-junta with one-sided error: we prove that for any $1 < k < n-1$, the query complexity is $\Omega(\log \binom{n}{k})$, which is almost optimal. This lower bound is a consequence of a proof that the query complexity of testing, with one-sided error, whether a function is a k-parity is $\Theta(\log \binom{n}{k})$.
2. Testing isomorphism between two unknown functions that can be queried: we prove that the query complexity in this setting is $\Omega(\sqrt{2^n})$ and $O(\sqrt{2^n n \log n})$.

1 Introduction

In this paper we address the following general question in the area of property testing:

QUESTION 1.1. *What is the query complexity of testing whether a black-box function $g : \{0,1\}^n \to \{0,1\}$ is isomorphic to a given function $f \in \mathcal{C}$, for various classes $\mathcal{C}$ of Boolean functions?*

This question is particularly interesting because testing many function properties, like those of being a dictatorship, a k-monomial, a k-parity and more, are equivalent to testing isomorphism to some function f. More general properties can often be reduced to testing isomorphism to several functions (as a simple example, notice that testing whether g depends on a single variable can be done by first testing if g is isomorphic to $f(x) \equiv x_1$, then testing if g is isomorphic to $f(x) \equiv 1 - x_1$, and accepting if one of the tests accepts). The "Testing by Implicit Learning" approach of Diakonikolas et al. [DLM+07] can also be viewed as a clever reduction from the task of testing a wide class of properties to testing function isomorphism against a number of functions.

On a wider perspective, an answer to Question 1.1 is an important step towards the meta-goal of characterizing testable properties of Boolean functions, as also suggested by [FKR+02] and [BO10].

There are several classes of functions for which testing isomorphism is trivial. For instance, if f is symmetric (invariant under permutations of variables), then testing f-isomorphism is equivalent to testing identity. More interesting functions are also known to have testers with constant query complexity. Specifically, the fact that isomorphism to dictatorship functions and k-monomials can be tested with $O(1)$ queries follows from the work of Parnas et al. [PRS02].

Fischer et al. [FKR+02] were the first to explicitly formulate the question of testing function isomorphism. They proved that isomorphism to any k-junta (function that depends on at most k variables) can be tested with roughly k^4 queries, whereas there are k-juntas for which testing isomorphism requires $\Omega(\log k)$ queries.[1] Motivated by problems in machine learning, the focus on juntas seems very natural in this context, especially due to the importance of dealing with functions on extremely large domains that depend only on few variables.

Combining the ideas from the testing algorithms of [FKR+02] with learning algorithms, Diakonikolas et al. [DLM+07] developed a powerful framework, called "Testing by Implicit Learning" for testing classes of functions that are well approximated by $O(1)$-juntas. Their results can be used to obtain isomorphism-testers for k-juntas as well, with query complexity roughly k^4 – similar to the one in [FKR+02]. We elaborate more

*Currently at Chennai Mathematics Institute, India; research performed while at CWI Amsterdam. Email: sourav@cmi.ac.in

†CWI Amsterdam. Email: david@cwi.nl

‡CWI Amsterdam. Email: arie.matsliah@gmail.com

[1] [FKR+02] actually prove more than that – a lower bound of $\Omega(\sqrt{k})$ for non-adaptive testers.

on [DLM+07] and how our work relates to it in the following section.

Blais and O'Donnell [BO10] proved query-complexity lower bounds for testing f-isomorphism for a wide class of functions. Specifically, [BO10] proved that testing isomorphism to *any* proper k-junta f (meaning that f is far from any $k-1$ junta) requires $\Omega(\log k)$ non-adaptive queries, which implies a general lower bound of $\Omega(\log\log k)$. In fact they establish the existence of a class of k-juntas such that testing isomorphism to any function in it requires $\Omega(\log k)$ non-adaptive (and $\Omega(\log\log k)$ adaptive) queries. They also proved that there is a k-junta (in particular, a majority on k variables) testing isomorphism to which requires $\Omega(k^{1/12})$ queries non-adaptively, and therefore $\Omega(\log k)$ queries in general.

Several related results, partially overlapping this work, were recently (and independently) obtained by Alon and Blais [AB10]. They proved that testing isomorphism non-adaptively to a known function requires $\Omega(n)$ queries in the worst case (a general lower bound of $\Omega(\log n)$ queries follows here too). With $k=n$, our lower bound is $\Omega(n)$ and it holds against adaptive testers as well. On the other hand, the lower bound in [AB10] is shown to hold for almost all functions $f : \{0,1\}^n \to \{0,1\}$, while our proof does not imply that. Alon and Blais also prove bounds similar to ours for the setting where both functions are unknown (see details in the next section).

2 Our results

2.1 Lower bounds for testing function isomorphism. Our first result (Theorem 6.1) is a lower bound of $\Omega(k)$, for any $1 \le k \le n$, on the query complexity of testing (adaptively, with two-sided error) isomorphism to k-juntas.

In fact, our proof yields a stronger result. To state it, let $\mathcal{F}_{\frac{n}{2}\pm\sqrt{n}}$ denote the set of all "truncated" functions $g : \{0,1\}^n \to \{0,1\}$ that satisfy $g(x) = 0$ for all x with $|x| \notin \frac{n}{2} \pm \sqrt{n}$; we prove the existence of k-juntas $f : \{0,1\}^n \to \{0,1\}$, for all $k \le n$, such that it is impossible to distinguish a random permutation of f from a random $g \in \mathcal{F}_{\frac{n}{2}\pm\sqrt{n}}$ with $o(k)$ queries. Furthermore, such an f can be quite restricted – it can be represented by a product of a threshold function and a polynomial over $\mathbb{F}_2$ of degree logarithmic in k; alternatively, it can be in (nonuniform) $\mathcal{NC}$.

As a corollary we obtain a lower bound of $\Omega(d)$ queries for testing if the Fourier degree of a Boolean function is at most d (see Theorem 7.2),[2] and a lower bound of $s^{\Omega(1)}$ queries for testing whether a function has a circuit of size s (see Theorem 7.1). These resolve open problems from [DLM+07].

We remark that the restriction of the foregoing indistinguishability result to truncated functions is essential – as Proposition 10 says (see Section 10), random permutations of *any* f can be distinguished from completely random functions with $\widetilde{O}(\sqrt{n})$ queries and arbitrarily high constant success probability (note that if the success probability is required to be only 3/4, a trivial such tester exists that makes only two queries: $\bar{0}$ and $\bar{1}$).

2.2 Upper bounds for testing function isomorphism. Our second result (Theorem 8.1) is a nearly matching upper bound of $O(k \log k)$ queries for testing isomorphism to any fixed k-junta. One consequence of our proof, which is of independent interest, is the following (see Proposition 8.2 for a formal statement):

Let $\epsilon > 0$ and suppose we are given oracle access to a k-junta $g : \{0,1\}^n \to \{0,1\}$. Then, after a preprocessing step that makes $O(k \log k)$ queries to g, we can draw uniformly random samples $(x, a) \in \{0,1\}^k \times \{0,1\}$ labelled by $\mathsf{core}(g) : \{0,1\}^k \to \{0,1\}$ – the "core" of g, such that for each sample (x, a), $\mathsf{core}(g)(x) = a$ with probability at least $1-\epsilon$. Furthermore, obtaining each sample requires making only one query to g.

Generating such samples is one of the main ingredients in the general framework of [DLM+07]; while the procedure therein makes $\Omega(k)$ queries to g for obtaining each sample (while executing k independence tests of Fischer et al. [FKR+02]), our procedure requires only *one* query to g per sample.

REMARK 2.1. *In a subsequent work (currently in preparation), we used a variation of this sampler to significantly improve the query-complexity of the testers from [DLM+07] for various Boolean function classes.*

2.3 Testing function isomorphism with one-sided error. Our third result (Theorem 5.1) concerns testing function isomorphism with one-sided error. The fact that the one-sided error case is strictly harder than the two-sided error case was proved by [FKR+02]. In particular, they showed the impossibility of testing isomorphism to 2-juntas with one-sided using a number of queries independent of n (their lower bound is $\Omega(\log\log n)$, which follows from an $\Omega(\log n)$ lower bound on non-adaptive testers). In this paper we prove nearly tight lower bounds for the problem. Specifically, we prove that the query complexity of testing isomorphism to k-juntas, for any $2 \le k \le n$, is between $\Omega(k \log(n/k))$ and $O(k \log n)$. (As we mentioned in the introduction,

[2] These bounds apply to the degree over *any* field. Better bounds are known for finite fields; c.f. [AKK+03, JPRZ04, KR04].

for $k = 1$ it can be done with $O(1)$ queries [PRS02].) The lower bound actually follows by the following result: the query complexity of testing (with one-sided error) whether a function is k-parity (i.e, an XOR of *exactly* k indices of its input), for any $2 \le k \le n-2$, is $\Theta(\log \binom{n}{k}) = \Theta(k \log(\frac{n}{\min\{k,n-k\}}))$. (In contrast, the well-known BLR test can test, with one-sided error, if a function is k-parity for *some* k using $O(1)$ queries).

2.4 Testing isomorphism between two unknown functions. Finally, we consider the related problem of testing isomorphism between two black-box functions (i.e., both f and g need to be queried). We show that the worst-case query complexity in this setting is between $\Omega(\sqrt{2^n})$ and $O(\sqrt{2^n n \log n})$. As mentioned in the introduction, similar results for this setting were independently obtained by Alon and Blais [AB10].

2.5 Summary.

In Table 1 we summarize our main results, and compare them to prior work. A few remarks are in order:

- Some of the lower bounds from prior work were obtained via exponentially larger lower bounds for non-adaptive testers, and some of them held only for limited values of k. The third column contains the details. Our lower bounds apply to general (adaptive, two-sided error) testers, and hold for all $k \le n$.
- The exponent in our $s^{\Omega(1)}$ bound for testing circuit size depends on the size of the smallest circuit that can generate s^4-wise independent distributions (see details in Section 7). In particular, standard textbook constructions show that the exponent is at least 1/8.
- The bounds in the last row have been independently and simultaneously obtained in [AB10].

We also comment that nearly all our results extend to functions with general product domains and general ranges, along the lines of [DLM$^+$07] and [Bla09].

Organization of this paper. After the necessary preliminaries, we give a brief overview of the main proofs in Section 4. The proofs for one-sided-error testing are given in Section 5. In Section 6 we present the lower bound on the query complexity of testing isomorphism to k-juntas, and the lower bound for testing the Fourier degree of a function. The lower bound for testing whether a function has a circuit of size s is given in Section 7. The algorithm for testing isomorphism to k-juntas is given in Section 8. In Section 9 we prove the bounds for testing isomorphism in the setting where both functions have to be queried. In Section 10 we prove that given any f, it is possible to distinguish, with high probability, a random permutation of f from a completely random function with $\widetilde{O}(\sqrt{n})$ queries.

3 Preliminaries

Most of our notation is quite standard or self explanatory; refer to Appendix A for clarification and for precise definitions of terms such as *k-junta*, *influence*, *relevant variable* and *property tester*. Here we only define the specific notation and terminology used throughout.

For $W \subseteq [n]$, we let $\{0,1\}^n_W$ denote the subset of $\{0,1\}^n$ containing strings with Hamming weight in W, namely, $\{0,1\}^n_W = \{x \in \{0,1\}^n : |x| \in W\}$. Additionally, let

$$\{0,1\}^n_{\frac{n}{2}\pm h} \triangleq \{x \in \{0,1\}^n : \frac{n}{2} - h \le |x| \le \frac{n}{2} + h\}.$$

Given a permutation $\pi \in Sym([n])$, there is a permutation $\phi(\pi) \in Sym(\{0,1\}^n)$ mapping $x = x_1 \dots x_n \in \{0,1\}^n$ to $(\phi(\pi))(x) \triangleq x_{\pi(1)} \dots x_{\pi(n)}$. (This is the natural action of π^{-1}). We will denote $\phi(\pi)$ itself also as π when no confusion is possible. We denote by $G_n \subseteq Sym(\{0,1\}^n)$ the image of ϕ; $|G_n| = n!$. Given a Boolean function f and $\pi \in G_n$, we write f^π for the function $f^\pi(x) \equiv f(\pi(x))$. In this notation, g is isomorphic to f, denoted by $g \cong f$, if and only if there is $\pi \in G_n$ such that $g \equiv f^\pi$.

The *distance up to permutations of variables*, denoted by $\mathrm{distiso}(f,g)$ is defined as $\min_\pi \mathrm{dist}(f^\pi, g)$. Testing f-isomorphism is the task of distinguishing the case $\mathrm{distiso}(f,g) = 0$ (or in short $f \cong g$) from the case $\mathrm{distiso}(f,g) \ge \epsilon$, with the objective of making as few queries to g as possible.

A function $f : \{0,1\}^n \to \{0,1\}$ is a k-parity if there is $v \in \{0,1\}^n$, $|v| = k$, such that $f(x) = \bigoplus_{i \in [n]} x_i v_i$ for all $x \in \{0,1\}^n$. The set of all k-parities is denoted PAR_k.

4 Brief overview of the main proofs

4.1 Overview of the lower bounds. The proof of Theorem 6.1 is done in two steps. First, we prove the existence of functions $f : \{0,1\}^n_{\frac{n}{2}\pm\sqrt{n}} \to \{0,1\}$ that are indistinguishable from random functions with fewer than roughly n queries. By this we mean that it is impossible to determine, with probability at least 2/3, whether g is a random permutation of f or a completely random function (on the domain $\{0,1\}^n_{\frac{n}{2}\pm\sqrt{n}}$), unless $\Omega(n)$ queries are made to g. Although it may seem that such an indistinguishability result might be obtained via straightforward probabilistic arguments, the actual

	prior work	non-adaptive	this work
testing isomorphism to k-juntas	$\Omega(\log k)$ [FKR+02, BO10, AB10] $\widetilde{O}(k^4)$ [FKR+02, DLM+07]	$\Omega(\sqrt{k})$ for $k \ll n$ [FKR+02] $\Omega(k^{1/12})$ for $k \ll n$ [BO10] $\Omega(k)$ for $k = n$ [AB10]	$\Omega(k)$ (Thm. 6.1) $O(k \log k)$ (Thm. 8.1)
testing isomorphism to k-juntas with 1-sided error	$\Omega(\log \log n)$ [FKR+02]	$\Omega(\log n)$ [FKR+02]	$\Omega(k \log (n/k))$ (Thm. 5.1) $O(k \log n)$ (Prop. 5.2)
testing the property of being a k-parity with 1-sided error			$\Theta(k \log(n/k))$ (Thm. 5.1)
testing if a function can be computed by a circuit of size s	$\widetilde{\Omega}(\log s)$ [DLM+07]		$s^{\Omega(1)}$ (Thm. 7.1)
testing if a function has Fourier degree $\leq d$	$\Omega(\log d)$ [DLM+07]	$\Omega(\sqrt{d})$ [DLM+07]	$\Omega(d)$ (Coro. 7.2)
testing isomorphism between two unknown functions	$\Omega(\sqrt{2^n}\ /n^{1/4})$ [AB10] $O(\sqrt{2^n\ n \log n/\epsilon})$ [AB10]		$\Omega(\sqrt{2^n})$ $O(\sqrt{2^n\ n \log n/\epsilon})$ (Thm. 9.1)

Table 1: Summary of results

proof has to overcome some technical difficulties. The main source of trouble is that we are dealing with a specific subset of permutations of $\{0,1\}^n$, induced by the set of permutations of $[n]$.

In the proof we borrow ideas from the work of Babai and Chakraborty [BC10], who proved query-complexity lower bounds for testing isomorphism of uniform hypergraphs. However, in order to be applicable to our problem, we have to extend the method of [BC10] in several ways. One of the main differences is that the permutation group in our case is not even transitive, which requires additional argument to prove that a random permutation "shuffles" the values of a function uniformly. Another difference is that for the proof of Theorem 7.1 we need a hard-to-test f that has a circuit of polynomial size, rather than just a random f. To address the second issue we relax the notion of uniformity from [BC10] to poly(n)-wise independence, and then apply standard partial derandomization techniques.

In the second step of the proof we show the $\Omega(k)$ lower bound for k-juntas by "padding" the hard-to-test functions from the previous step. The main argument in this part of the proof is showing that for any $f, g : \{0,1\}^k \to \{0,1\}$ and their extensions (paddings) $f', g' : \{0,1\}^n \to \{0,1\}$, $\text{distiso}(f', g') = \Omega(\text{distiso}(f, g))$. (Notice that an exact equality between the two distances does not hold; consider e.g. the functions $f(x) \equiv |x| \bmod 2$ and $g(x) \equiv 1 - f(x)$).

4.2 Overview of the upper bounds. The main ingredient in the proof of Theorem 8.1 is the nearly-optimal junta tester of Blais [Bla09]. In fact, a significant part of our proof deals with analyzing the junta tester of Blais, and proving that it satisfies stronger conditions than what was required for the mere task of junta-testing.

Let us briefly describe the resulting isomorphism tester: The algorithm begins by calling the junta tester, which may either reject (meaning that g is not a k-junta), or otherwise provide a set of $k' \leq k$ blocks (subsets of indices) such that if g is close to some k'-junta h', then with high probability, h' has at most one relevant variable in each of the k' blocks. Using these k' blocks we define an extension h of h' (if $k' < k$), and a noisy sampler S that provides random samples $(x, a) \in \{0,1\}^k \times \{0,1\}$, such that $\Pr[h(x) \neq a]$ is sufficiently small. Finally, we use the (possibly correlated) noisy samples of S to test if h is $\epsilon/10$-close to the core function of f or $9\epsilon/10$-far from it.

We note that our approach resembles the high-level idea in the powerful "Testing by Implicit Learning" paradigm of Diakonikolas et al. [DLM+07]. Furthermore, an upper bound of roughly k^4 queries to our problem follows easily from the general algorithm of [DLM+07]. (It seems that using the recent results of [Bla09], the algorithm of [DLM+07] can give an upper bound of roughly k^3.)

Apart from addressing a less general problem, there are several additional reasons why our algorithm attains a tighter upper bound of $k \log k$. First, in our case the known function is a proper junta, and not just approximated by one. Second, while simulating random samples from the core of the unknown function g, we allow a small, possibly correlated, fraction of the samples to be incorrectly labelled. This enables us to generate a random sample with just one query to g, sparing us the need to perform the Independence-Tests of [FKR+02]. Then we perform the final test (the parallel of Occam's razor from [DLM+07]) with a tester that is tolerant (i.e. it accepts even if the distance to the defined property is small) and resistant against (possibly correlated) noise.

4.3 Overview of the lower bound for testing with one-sided error. As mentioned earlier, the lower bound, which is the interesting part of Theorem 5.1, is obtained via a lower bound for testing isomorphism to k-parities with one-sided error.

We start with the simple observation that testing isomorphism to k-parities is equivalent to testing isomorphism to $(n-k)$-parities. Since testing 0-parities (constant zero functions) takes $O(1)$ queries, and testing 1-parities (dictatorship functions) takes $O(1)$ queries as well (by Parnas et al. [PRS02]), we are left with the range $2 \le k \le n/2$.

We split this range into three parts: small (constant) k, medium k and large k. For small k's a lower bound of $\Omega(\log n)$ is quite straightforward. For the other two ranges, we use the Frankl–Wilson and Frankl–Rödl theorems, which bound the size of families of subsets with restricted intersection sizes. (The reason for this case distinction is to comply with the hypotheses of the combinatorial theorems). We obtain lower bounds of $\Omega(k \log(n/k))$ and $\Omega(k)$, respectively.

In all three cases we follow the same argument: suppose that we want to prove a lower bound of $q = q(n,k)$. We define a function g that is either a k'-parity (for a suitably chosen $k' \neq k$) [3] or a constant, and depends only on n and k. This function has the property that for all $x^1, \ldots, x^q \in \{0,1\}^n$ there exists a k-parity f satisfying $f(x^i) = g(x^i)$ for all $i \in [q]$. This forces any one-sided error tester making $\le q$ queries to accept g, even though it is 1/2-far from any k-parity.

5 Testing function isomorphism with one-sided error

THEOREM 5.1. *The query complexity of testing isomorphism to k-juntas with one-sided error is between* $\Omega(k \log \frac{n}{k})$ *and* $O(k \log n)$.

Note that if $f \in \mathsf{PAR}_k$, then testing isomorphism to f is the same as testing membership in PAR_k. Hence the lower bound in Theorem 5.1 follows from the next proposition.

PROPOSITION 5.1. *Let* $\epsilon \in (0, \frac{1}{2}]$ *be fixed. The following holds for all* $n \in \mathbb{N}$*:*

- *For any* $k \in [2, n-2]$*, the query complexity of testing* PAR_k *with one-sided error is* $\Theta(\log \binom{n}{k})$*. Furthermore, the upper bound is obtainable with a non-adaptive tester, while the lower bound applies to adaptive tests, and even to the certificate size for proving membership in* $\overline{\mathsf{PAR}_k}$*.*
- *For any* $k \in \{0, 1, n-1, n\}$*, the query complexity of testing* PAR_k *with one-sided error is* $\Theta(1)$*.*

For every $f : \{0,1\}^n \to \{0,1\}$ let Isom_f denote the set of functions isomorphic to f. The upper bound in Theorem 5.1 follows from the next proposition.

PROPOSITION 5.2. *Isomorphism to any given* $f : \{0,1\}^n \to \{0,1\}$ *can be tested with* $O(\log |\mathsf{Isom}_f|/\epsilon)$ *queries.*

This immediately implies the desired upper bound, since $|\mathsf{Isom}_f| \le \binom{n}{k} \cdot k!$ for any $k \in [n]$ and k-junta f. This also implies the upper bound in the first item of Proposition 5.1, since for a k-parity f, $|\mathsf{Isom}_f| = |\mathsf{PAR}_k| = \binom{n}{k}$.

5.1 Proof of Proposition 5.1. We begin with the following observation, which is immediate from the fact that p is a k-parity if and only if $p(x) \oplus x_1 \oplus \ldots \oplus x_n$ is an $(n-k)$-parity:

OBSERVATION 5.1. *Let* $\epsilon \in (0, \frac{1}{2}]$*,* $n \in \mathbb{N}$ *and* $k \in [0, n]$*. Any* ϵ*-tester for* PAR_k *can be converted into an* ϵ*-tester for* PAR_{n-k}*, while preserving the same query complexity, type of error, and adaptivity.*

As mentioned earlier, the upper bound in the first item of Proposition 5.1 follows by Proposition 5.2. It is also easy to verify that the second item holds for $k = 0$. For $k = 1$, the bound follows from [PRS02], who show that one-sided-error testing of functions for being a 1-parity (monotone dictatorship) can be done with $O(1)$ queries. So, according to Observation 5.1 we only have to prove the lower bound in the first item of Proposition 5.1 for $k \in [2, \lfloor n/2 \rfloor]$.

To this end we make a distinction between three cases. First we prove a lower bound of $\Omega(\log n)$ for any $k \in [2, \lfloor n/2 \rfloor]$. Then a lower bound of $\Omega(\log \binom{n}{k})$ is shown for $k \in [5, \alpha n]$, where $\alpha n \triangleq \lfloor n/2^{12} \rfloor$. Finally we prove a lower bound of $\Omega(k)$ queries that works for $k \in [\alpha n, \lfloor n/2 \rfloor]$. Combining the three bounds will complete the proof.

In all three cases we follow the argument sketched in the overview (Section 4.3).

5.1.1 Lower bound of $\Omega(\log n)$ for $2 \le k \le \lfloor n/2 \rfloor$. Let $q = \lfloor \log n \rfloor - 1$, and let $x^1, \ldots, x^q \in \{0,1\}^n$ be the set of queries. For any $k \in [2, \lfloor n/2 \rfloor]$ we let g be the $(k-2)$-parity $g(x) = x_{n-k+3} \oplus \cdots \oplus x_n$ (in case $k = 2$, g is simply the constant zero function). Then we find $j, j' \in [n-k+2]$, $j \neq j'$ such that $x^i_j = x^i_{j'}$ for all $i \in [q]$;

[3] Note that not every choice of k' works, even if k and k' are very close to each other. For example, if $k' = k + 1$, it is easy to tell PAR_k from $\mathsf{PAR}_{k'}$ by simply querying the all-ones vector.

such j and j' must exist since $2^q < n-k+2$. Let f be the k-parity corresponding to $\{j,j'\} \cup [n-k+3,n]$. Then $f(x^i) = g(x^i)$ for all $i \in [q]$, so the tester must accept g, even though it is 1/2-far from any k-parity.

This simple idea can only yield lower bounds of $\Omega(\log n)$. We need to generalize it in order to obtain lower bounds that grow with k.

5.1.2 Lower bound of $\Omega(\log\binom{n}{k})$ for $5 \le k \le \alpha n$.

For this case we use the following version of the Frankl-Wilson Theorem:

Theorem 5.2. ([FW81]; see also [FR87]) *Let $m \in \mathbb{N}$ and let $\ell \in [m]$ be even, such that $\ell/2$ is prime power. If $\mathcal{F} \subseteq \binom{[m]}{\ell}$ is such that for all $F, F' \in \mathcal{F}$, $|F \cap F'| \ne \ell/2$, then $|\mathcal{F}| \le \binom{m}{\ell/2}\binom{3\ell/2-1}{\ell}/\binom{3\ell/2-1}{\ell/2}$.*

Let $q = \lfloor \frac{1}{20}\log\binom{n}{k}\rfloor$. Given $k \in [5, \lfloor n/2\rfloor]$, let $k' \ge 1$ be the smallest integer such that $(k-k')/2$ is a power of a prime; note that $k' < k/2$ as $k \ge 5$. We let g be the k'-parity $g(x) = x_{n-k'+1} \oplus \cdots \oplus x_n$. With a slight abuse of notation, let g also denote the n-bit string with ones exactly in the last k' indices. It suffices to show that for any $x^1,\dots,x^q \in \{0,1\}^n$ there exists $y \in \{0,1\}^n$ such that

- $|y| = k - k'$,
- $y \cap g = \emptyset$ and
- $\langle y, x^i\rangle \triangleq \bigoplus_{j=1}^n (y_j \cdot x^i_j) = 0$ for all $i \in [q]$.

Indeed, if such a y exists, then the k-parity corresponding to $g \cup y$ is consistent with g on $x^1,\dots,x^q$.

Let $Y = \{y \in \{0,1\}^n : |y| = k-k' \text{ and } y \cap g = \emptyset\}$. Partition Y into disjoint subsets $\{Y_\alpha\}_{\alpha\in\{0,1\}^q}$, such that $y \in Y_\alpha$ if and only if $\langle y, x^i\rangle = \alpha_i$ for all $i \in [q]$. Clearly, one of the sets Y_α must be of size at least $\binom{n-k'}{k-k'}/2^q$. We interpret the elements of this Y_α as ℓ-subsets of $[m]$, where $\ell \triangleq k-k'$ and $m \triangleq n-k'$, and show that there must be $y^1, y^2 \in Y_\alpha$ such that $|y^1 \cap y^2| = \ell/2 = (k-k')/2$. Once the existence of such a pair is established, the claim will follow by taking y to be the bitwise XOR of y^1 and y^2. Indeed, it is clear that $|y| = k-k'$ and $y \cap g = \emptyset$, and it is also easy to verify that $\langle y, x^i\rangle = \langle y^1, x^i\rangle \oplus \langle y^2, x_i\rangle = 0$ for all $i \in [q]$.

Let $c \triangleq n/k$; observe that $c \le m/\ell \le 2c$. In the following we use the bounds $b(\log(a/b)) \le \log\binom{a}{b} \le b(\log(a/b)+2)$.

We have

$$\begin{aligned}
\log|Y_\alpha| &\ge \log\left(\frac{\binom{n-k'}{k-k'}}{2^q}\right) \ge \log\binom{m}{\ell} - \frac{1}{20}\log\binom{n}{k} \\
&\ge \ell(\log(m/\ell)) - \frac{1}{20}k(\log(n/k)+2) \\
&\ge \ell(\log c) - \frac{1}{10}\ell(\log c + 2) \\
&= \ell\left(\frac{9}{10}\log c - \frac{1}{5}\right).
\end{aligned}$$

On the other hand,

$$\begin{aligned}
\log\left(\frac{\binom{m}{\ell/2}\binom{3\ell/2-1}{\ell}}{\binom{3\ell/2-1}{\ell/2}}\right) &\le \frac{\ell}{2}(\log(m/\ell)+3) + 3\ell/2 \\
&\le \ell\left(\frac{1}{2}\log c + \frac{7}{2}\right).
\end{aligned}$$

Since $c \ge 2^{12}$, these inequalities together with Theorem 5.2 imply that there must be $y^1, y^2 \in Y_\alpha$ such that $|y^1 \cap y^2| = \ell/2$, as desired.

5.1.3 Lower bound of $\Omega(k)$ for $\alpha n \le k \le \lfloor n/2\rfloor$.

The reasoning in this case is very similar, but since for large k the previous method does not work, we have to change few things. One of them is switching to the related theorem of Frankl and Rödl, using which we can prove a lower bound of $\Omega(k)$ (instead $\Omega(\log\binom{n}{k})$), but for the current range of k they are asymptotically the same.

Theorem 5.3. ([FR87], Thm. 1.9) *There is an absolute constant $\delta > 0$ such that for any even k the following holds: Let $\mathcal{F}$ be a family of subsets of $[2k]$ such that no two sets in the family have intersection of size $k/2$. Then $|\mathcal{F}| \le 2^{(1-\delta)2k}$.*

Let n be large enough with respect to α and δ. Given $k \in [\alpha n, \lfloor n/2\rfloor]$, we set $q = \delta k$. Assume first that k is even – we mention the additional changes required for odd k below.

We set g to be the zero function, and show that for any $x^1,\dots,x^q \in \{0,1\}^n$ there exists $y \in \{0,1\}^n$ such that

- $|y| = k$ and
- $\langle y, x^i\rangle = 0$ for all $i \in [q]$.

Let $Y = \{y \in \{0,1\}^n : y \subseteq [2k] \text{ and } |y| = k\}$. As in the previous case, partition Y into disjoint subsets $\{Y_\alpha\}_{\alpha\in\{0,1\}^q}$, such that $y \in Y_\alpha$ if and only if $\langle y, x^i\rangle = \alpha_i$ for all $i \in [q]$. One of the sets Y_α must be of size at least $\binom{2k}{k}/2^q = 2^{2k-1-q}$, which is greater than $2^{(1-\delta)2k}$ for large enough n (and hence k). We interpret the elements of this Y_α as k-subsets of $[2k]$ in the natural way. Thus, by Theorem 5.3, there must be $y^1, y^2 \in Y_\alpha$ such that $|y^1 \cap y^2| = k/2$. Take y to be the bitwise XOR of y^1 and y^2. Clearly $|y| = k$, and $\langle y, x^i \rangle = 0$ for all $i \in [q]$.

For an odd k, we use the 1-parity $g(x) = x_n$ instead of the zero function. We follow the same steps to find $y \subseteq [2k-2]$ of size $|y| = k-1$ such that $\langle y, x^i \rangle = 0$ for all $i \in [q]$. Then, the vector $y \cup \{n\}$ corresponds to a function in PAR_k that is consistent with g on the q queries.

5.2 Proof of Proposition 5.2 Consider the simple tester described in Algorithm 1. It is clear that this is a non-adaptive one-sided error tester, and that it only makes $O(\log|\mathsf{Isom}_f|/\epsilon)$ queries to g. So we only need to show that for any f and any g that is ϵ-far from f, the probability of acceptance is small. Indeed, for a fixed $h \in \mathsf{Isom}_f$ the probability that $g(x^i) = h(x^i)$ for all $i \in [q]$ is at most $(1-\epsilon)^q$. Applying the union bound on all functions $h \in \mathsf{Isom}_f$, we can bound the probability of acceptance by $|\mathsf{Isom}_f|(1-\epsilon)^q \le |\mathsf{Isom}_f|e^{-\epsilon q} < 1/3$. □

Algorithm 1 (non-adaptive one-sided error tester for the known-unknown setting)

let $q \leftarrow \frac{2}{\epsilon}\log|\mathsf{Isom}_f|$
for $i = 1$ to q **do**
  pick $x^i \in \{0,1\}^n$ uniformly at random
  query g on x^i
end for
accept if and only if there exists $h \in \mathsf{Isom}_f$ such that $g(x^i) = h(x^i)$ for all $i \in [q]$

6 $\Omega(k)$ lower bound for testing isomorphism to k-juntas

Definition 6.1. *Let $\mathcal{F}_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$ denote the set of all "truncated" functions $g : \{0,1\}^n \to \{0,1\}$, i.e. those satisfying $g(x) = 0$ for all x with $|x| \ne \frac{n}{2} \pm \lceil\sqrt{n}\rceil$; a* random truncated function *is a random function uniformly drawn from $\mathcal{F}_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$.*

Observation 6.1. *Let $f : \{0,1\}^n \to \{0,1\}$ be any function and $g : \{0,1\}^n \to \{0,1\}$ a random truncated function. Let ϵ_0 be an arbitrary constant. Then, with probability $1 - o(1)$ over the choice of g, $\mathrm{distiso}(f,g) \ge \epsilon_0$.*

It follows that, in order to prove lower bounds for testing isomorphism to f, it suffices to show the stronger claim that there is a function g such that no algorithm is able to distinguish between a random permutation of f and a random permutation of g:

Definition 6.2. *Let $f, g : D \to \{0,1\}$ be Boolean functions on some domain D and $\epsilon > 0$. We say that the pair (f,g) is (q,ϵ)-hard if $\mathrm{distiso}(f,g) \ge \epsilon$ and, given oracle access to a function h that is promised to be a random permutation of f or g (each with probability half), it is impossible to determine which is the case with overall probability $\ge 2/3$ unless q queries are made.*

We can rephrase the previous observation as stating that the existence of an q-hard pair f, g implies a lower bound of q on the query complexity of testing isomorphism to f (or g, for that matter).

Theorem 6.1. *There is some $\epsilon > 0$ such that for any $k \le n$ there is a k-junta $f : \{0,1\}^n \to \{0,1\}$ with the property that, for most truncated functions $g : \{0,1\}^n \to \{0,1\}$, the pair (f,g) is $(k - 5\log k, \epsilon)$-hard. Moreover, f can have either one of the following properties:*

- *f can be written as a product of two threshold functions and a polynomial of degree $O(\log k)$*
- *f is in $\mathcal{NC}$, i.e. it can be computed by circuits of size $\mathrm{poly}(k)$ and depth $O(\mathrm{polylog}(k))$.*

The proof of Theorem 6.1 is done in three steps:

1. In the first (and main) step we show that there is an $(\Omega(k), \Theta(1))$-hard pair of "nice" functions $f, g : \{0,1\}^k_{k/2\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$. This is proved in Proposition 6.1, which is the main technical result of this section.
2. In the second step we show that there is an $(\Omega(k), \Theta(1))$ pair of "nice" *truncated* functions $f, g : \{0,1\}^k \to \{0,1\}$. This is proved in Corollary 6.1, which follows easily from Proposition 6.1: we extend the function obtained in step one to the whole cube, by assigning zeroes outside the middle layers; then the claim follows by observing that inputs in the middle layers constitute a constant fraction of all inputs, and that the zeroes outside the middle layers cannot help in the testing process. As we mentioned in Section 2.1 in the introduction, this step is essential (see Section 10 for a formal proof).
3. The last step in the proof of Theorem 6.1 uses a "preservation of distance under padding" argument (Lemma 6.1 below), which essentially allows us to embed a function on k variables into one on n variables, so that testing hardness remains roughly the same.

6.1 The three steps and the proof of Theorem 6.1.

Proposition 6.1. (the main step) *For any k there is a function $f : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$ such that, with probability $1-o(1)$ over the choice of a random truncated g, pair (f,g) is is $(k - 5\log k)-$hard. Moreover, f can have either of the following properties:*

- *f can be evaluated by a degree $4\log k$ polynomial over $\mathbb{F}_2$,*
- *f can be computed by an $\mathcal{NC}$ circuit.*

The proof is deferred to Section 6.2.

Corollary 6.1. (extending $\{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}$ to $\{0,1\}^k$) *For any k, there is a function $f : \{0,1\}^k \to \{0,1\}$ such that, with probability $1 - o(1)$ over the choice of a random truncated function $g : \{0,1\}^k \to \{0,1\}$, pair (f,g) is $(k - 5\log k, \Theta(1))$-hard. Moreover, f can have either of the following properties:*

- *f can be written as a product of two threshold functions and a polynomial of degree $O(\log k)$,*
- *f can be computed by a $\mathcal{NC}$ circuit.*

Proof. For a function $h : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$, call $ext(h) : \{0,1\}^k \to \{0,1\}$ the extension of h to the whole of $\{0,1\}^k$ vanishing outside $\{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}$. Observe that a random truncated function is nothing more than the extension of a random $g : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}$. Also note that for any pair f, g, the equality $\text{distiso}(ext(f), ext(g)) = \alpha\ \text{distiso}(f,g)$ holds, where $\alpha \triangleq |\{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}|/2^k = \Theta(1)$.

Let ϵ_0 be as in Observation 6.1; then a random truncated g satisfies $\text{distiso}(f,g) \geq \epsilon_0$ with probability $1 - o(1)$; fix any such g. Any algorithm $\mathcal{A}(h')$ distinguishing between random permutations of $ext(f)$ and $ext(g)$ when allowed query access to h' can be turned into one distinguishing permutations of f and g in the obvious way: given access to $h : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$, run $\mathcal{A}(ext(h))$. This works because $h = f^\pi \leftrightarrow ext(h) = ext(f)^\pi$. This shows that distinguishing between random permutations of $ext(f)$ and $ext(g)$ is as hard as distinguishing between random permutations of f and g, which by hypothesis requires q queries. Since we also have $\text{distiso}(ext(f), ext(g)) = \alpha\ \text{distiso}(f,g) \leq \epsilon$, where $\epsilon \triangleq \alpha\epsilon_0$, we conclude that $(ext(f), ext(g))$ is (q,ϵ)-hard.

By Proposition 6.1, we have $f : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$ that can either be represented by a degree $O(\log k)$ polynomial or by a $\mathcal{NC}$ circuit. All that remains to be shown is that $ext(f)$ has the required simplicity property; for this we need to compose f with threshold functions. More specifically, let $A, B : \{0,1\}^k \to \{0,1\}$ be given by $A(x) = 1$ iff $|x| \geq k/2 - \lceil\sqrt{k}\rceil$ and $B(x) = 1$ iff $|x| \leq k/2 + \lceil\sqrt{k}\rceil$. It is well known that $A, B \in \mathcal{NC}$, so the function $f'(x) = f(x) \wedge A(x) \wedge B(x)$ has the desired properties. □

Lemma 6.1. (extension from $\{0,1\}^k$ to $\{0,1\}^n$) *Let $k, n \in \mathbb{N}$, $k \leq n$, and let $f', g' : \{0,1\}^k \to \{0,1\}$ be a pair of functions. Define $f = pad(f')$ to be the padding extension of f', where $f : \{0,1\}^n \to \{0,1\}$ is given by $f(x) = f'(x\restriction_{[k]})$ for all $x \in \{0,1\}^n$. Likewise, define $g = pad(g')$. Then the following holds:*

- $\text{distiso}(f,g) \geq \text{distiso}(f',g')/3$.
- *If (f',g') is (q,ϵ)-hard, then (f,g) is $(q,\epsilon/3)$-hard.*

Proof. Let $\epsilon \triangleq 3 \cdot \text{distiso}(f,g)$; then $\text{dist}(f, g^\pi) = \epsilon/3$ for some permutation π. Write $A \triangleq \pi([k]) - [k]$, and let $I(A) \triangleq Inf_{g^\pi}(A)$ denote the influence A on g^π. First observe that $I(A) \leq 2 \cdot \text{dist}(f, g^\pi) \leq 2\epsilon/3$, due to the fact that f does not depend on the indices in A at all. This is easy to verify directly; see also Lemma 8.2.

Let $\sigma : A \to \Big([k] \setminus \pi([k])\Big)$ be an arbitrary bijection. Consider the permutation $\pi' : [k] \to [k]$ defined as

$$\pi'(i) = \begin{cases} \pi(i) & , \quad i \in [k] \text{ and } \pi(i) \in [k] \\ \sigma(\pi(i)) & , \quad i \in [k] \text{ and } \pi(i) \in A \end{cases}$$

Informally, π' is obtained from π by "bringing back" to $[k]$ all those $i \in [k]$ that were mapped to A. We have $\text{dist}(g^\pi, g^{\pi'}) \leq I(A) \leq 2\epsilon/3$, and by the triangle inequality, $\text{dist}(f, g^{\pi'}) \leq \epsilon$. Hence $\text{distiso}(f',g') \leq \text{dist}(f', g'^{\pi'}) = \text{dist}(f, g^{\pi'}) \leq \epsilon$, equality due to π' defining a valid permutation of $[k]$.

It is clear that if $f' \cong g'$ then $f \cong g$. Let there be an algorithm $\mathcal{A}$ capable of distinguishing a random permutation of f from a random permutation of g using fewer than q queries. Based on $\mathcal{A}$, we can construct an algorithm to distinguish whether $h' : \{0,1\}^k \to \{0,1\}$ is a random permutation of f' or a random permutation of g' in the following manner: pick a uniformly random permutation $\sigma \in Sym([n])$, and apply $\mathcal{A}$ to $pad(h)^\sigma$ (clearly, any query to h' can be simulated by one query to $pad(h')^\sigma$, and the distribution of $pad(h')^\sigma$ is a random permutation of either f or g). Hence no such $\mathcal{A}$ exists. □

Combining Corollary 6.1 and Lemma 6.1 yields Theorem 6.1. The rest of this section is devoted to the proof of Proposition 6.1.

6.2 Proof of Proposition 6.1.

remark 6.1. *For notational convenience and compatibility with external lemmas, we replace k with n throughout the proof of Proposition 6.1.*

6.2.1 Central lemmas.

In the following, the notation $a = (1 \pm b)c$ will be understood to mean $(1-b)c \le a \le (1+b)c$.

Definition 6.3. *Let T be a finite domain and $r \in \mathbb{N}$. We say that a* ***multiset*** *$\mathcal{F}$ of functions from T to $\{0,1\}$ is* r-uniform *(with regard to a group G of permutations of T) if*

- *$\mathcal{F}$ is closed under the action of G: for all $\pi \in G$ and $f \in \mathcal{F}$, $f^\pi \in \mathcal{F}$.*
- *$\mathcal{F}$ is r-independent: for all $Q \in \binom{T}{r}$ and $a : Q \to \{0,1\}$, $\Pr_{f\in\mathcal{F}}[f\restriction_Q = a] = 2^{-r}$.*

In this section we will always take $G = G_n$ to be the "permutation of variables" subgroup of $Sym(\{0,1\}^n)$ defined in the preliminaries. As an example, the family of all Boolean functions on T is $|T|$-uniform with regard to G.

Definition 6.4. *Let $\delta \in \mathbb{R}^+, q \in \mathbb{N}$. We say that a Boolean function $f : T \to \{0,1\}$ is* (q,δ)-regular *if for all $Q \in \binom{T}{q}$ and $a : Q \to \{0,1\}$*

$$\Pr_{\tau\in G}[f^\tau\restriction_Q = a] = (1\pm\delta)2^{-q},$$

That is, the probability in question is close to the probability that a random Boolean function on Q coincides with a. The idea is that two functions that are both regular will be hard to tell from each other.

Lemma 6.2. *Let $\delta > 0$ be a constant, $N \triangleq \binom{n}{n/2-\lceil\sqrt{n}\rceil}$ and $\mathcal{F}$ be an r-uniform family of Boolean functions on $\{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$. If $q = \log N - 5\lceil\log n\rceil$ and $r = n^4$, then a random function from $\mathcal{F}$ is (q,δ)-regular with probability $1-o(1)$.*

Proof. Fix $Q \in \binom{T}{q}$ (where, T denotes $\{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$) and $a : Q \to \{0,1\}$. For any $g : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$ and $\tau \in G$, define the indicator variable $X(g,\tau) = \mathbb{I}[g^\tau\restriction_Q = a]$. Define $A(f) \triangleq \Pr_{\tau\in G}[X(f,\tau) = 1]$; we aim to compute the probability, over random f, that $A(f)$ deviates from $p = 1/2^q$ by more than δp. Notice that $\mathbb{E}_f[A(f)] = \mathbb{E}_\tau \mathbb{E}_f X(f,\tau) = \mathbb{E}_\tau p = p$, where we made use of uniformity of $\mathcal{F}$ and the fact that $r \ge q$.

Consider any pair $\sigma_1, \sigma_2 \in G$ such that $\sigma_1(Q) \cap \sigma_2(Q) = \emptyset$. Since $2q \le r$, a random function from $\mathcal{F}$ assigns values independently on each element of $\sigma_1(Q) \cup \sigma_2(Q)$, so the random variables $X(f,\sigma_1)$ and $X(f,\sigma_2)$ are independent conditioned on the choice of σ_1,σ_2.

More generally, for any s permutations $\sigma_1,\ldots,\sigma_s$ of G under which the images of Q are pairwise disjoint, and for any $\pi \in G$, the variables $X(f,\pi\circ\sigma_1),\ldots,X(f,\pi\circ\sigma_s)$ are n^3-wise independent, since $r \ge n^3 q$. They are also uniform because the distributions of f and $f^{\pi\circ\sigma_i}$ are the same for f drawn from $\mathcal{F}$. We will need a large set of permutations with this property:

Lemma 6.3. *There exist $s \triangleq \lceil N/q^2\rceil$ permutations $\sigma_1,\ldots,\sigma_s \in G$ such that $\sigma_1 Q,\ldots,\sigma_s Q$ are disjoint.*

Proof. First note that for any $x, y \in \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$,

$$\Pr_{\pi\in G}[\pi x = y] = \left\{\begin{array}{ll} 0, & |x| \ne |y| \\ \frac{1}{\binom{n}{|x|}}, & |x| = |y| \end{array}\right\} \le \frac{1}{N}.$$

This holds because the orbit of x under G is the set of all $\binom{n}{|x|}$ strings of the same weight.

Let $\Sigma \subseteq G$ be a maximal set of permutations satisfying the hypothesis of the lemma; write $s \triangleq |\Sigma|$ and $V = \bigcup_{\sigma\in\Sigma}\sigma Q$. Then $|V| = qs$ and maximality means that every πQ has non-empty intersection with V. Therefore $1 = \Pr_{\pi\in G}[\exists x \in Q, y \in V$ such that $\pi x = y] \le \frac{q^2 s}{N}$, where we used the the union bound over x and y. Thus $s \ge \frac{N}{q^2}$. $\square$

For any $\pi \in G$, $A(f) = A(f^\pi) = \mathbb{E}_{\tau\in G} X(f,\tau\circ\pi)$. In particular, drawing π from $\sigma_1,\ldots,\sigma_s$ at random, $A(f)$ also equals the average value

$$\begin{aligned} A(f) &= \mathop{\mathbb{E}}_{i\in[s]}\mathop{\mathbb{E}}_{\tau\in G} X(f,\tau\circ\sigma_i) \\ &= \mathop{\mathbb{E}}_{\tau\in G}\mathop{\mathbb{E}}_{i\in[s]} X(f,\tau\circ\sigma_i) = \mathop{\mathbb{E}}_{\tau} Y(f,\tau), \end{aligned}$$

where $Y(f,\tau) = \mathbb{E}_i X(f,\tau\circ\sigma_i)$. We need to show that for typical f, $\mathbb{E}_\tau Y(f,\tau)$ is close to p; clearly it suffices to prove that $\delta \triangleq \max_\tau |Y(f,\tau) - p|$ is small for such f.

When τ is fixed, $Y(f,\tau)$ is the average of s k-wise independent random variables (with $k \triangleq n^3$), each satisfying $\mathbb{E}_f X(f,\tau\circ\sigma_i) = p$. We will need the following version of Chernoff bounds:

Lemma 6.4. (Chernoff bounds for k-wise indep.) *[SSS95] Let X be the sum of s k-wise independent random variables in the interval $[0,1]$, and let $p = \frac{1}{s}\mathbb{E}[X]$. For any $0 \le \delta \le 1$,*

$$\Pr[|X - p| \ge \delta p] \le e^{-\Omega(\min(k,\delta^2 ps))}.$$

Since $ps \ge n^3$ and $k = n^3$, using Lemma 6.4 we obtain

$$\forall\tau\ \Pr_f[|Y(f,\tau) - p| > p\delta] = 2^{-\Omega(\delta n^3)},$$

hence we can upper bound $\Pr_f[|A(f) - p| > p\delta]$ by

$$\Pr_f[\exists\tau \in G : |Y(f,\tau) - p| > p\delta] \le |G|2^{-\Omega(\delta n^3)}.$$

To conclude, we apply the union bound again, this time over all possible choices of Q and $a \in \{0,1\}^Q$, yielding

$$\Pr_f[\exists\, Q, a : |A(f) - p| > p/5] \leq \binom{2^n}{q} 2^q n! 2^{-\Omega(\delta n^3)},$$

which is $o(1)$.

□

6.2.2 Proof of the main claim of Proposition 6.1. We first prove the existence of a function $f : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$ satisfying all conditions except the last two items on the "niceness" of f.

Let $q \triangleq n - 5\log n$. Take two random functions $f, g : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$, with f drawn from a n^4-uniform family and g uniformly random. With probability $1 - o(1)$, $\text{distiso}(f, g) = \Omega(1)$. Also, by Lemma 6.2, both functions are (q, δ)-regular, where we picked some $\delta < 1/5$.

Consider the following two distributions:

- D_Y: pick $\pi \in G$ uniformly at random, and return f^π.
- D_N: pick $\pi \in G$ uniformly at random, and return g^π.

By definition, any $y \in D_Y$ is isomorphic to f, whereas any $n \in D_n$ is $\text{distiso}(f, g)$-far from it (and isomorphic to g). Let h be in the support of D_Y or D_N. Then h is also (r, δ)-regular, implying that for any $Q \in \binom{T}{q}$ (where, $T \triangleq \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$) and $a : Q \to \{0,1\}$,

$$\frac{4}{5 \cdot 2^q} < \Pr_\pi[h^\pi\restriction_Q = a] < \frac{6}{5 \cdot 2^q},$$

so $(2/3)Pr_{y\in D_Y}[y\restriction_Q = a] < Pr_{n\in D_N}[n\restriction_Q = a]$ and an appeal to Lemma A.1 establishes the main claim. Next we prove the two items in Proposition 6.1.

6.2.3 Proof of item 1 of Proposition 6.1. We need the following lemma, which gives us a n^4-uniform family of functions to draw f from, which is all required at this point to establish item 1.

Lemma 6.5. *Let $\mathcal{F}_d$ be the set of all polynomials $p : \mathbb{F}_2^n \to \mathbb{F}_2$ of degree at most d. Then $\mathcal{F}_d$ is $(2^{d+1} - 1)$-uniform.*

Proof. $\mathcal{F}_d$ is obviously closed under permutations of variables. With regard to independence, is enough to prove the following claim: for any set $S \subseteq F_2^n$ of size $|S| < 2^{d+1}$, and any function $f : S \to F_2$, there is a polynomial $q \in \mathcal{F}_d$ such that $q\restriction_S = f$; this fact has been generalized in the works of [KS05] and [BEHL09]. Indeed, if the claim holds then $\Pr_{p\in\mathcal{F}_d}[p\restriction_S = f] = \Pr_{p\in\mathcal{F}_d}[(p \oplus q)\restriction_S = 0] = \Pr_{p'\in\mathcal{F}_d}[p'\restriction_S = 0]$, since the distributions of p and $p' \triangleq p \oplus q$ are uniform over $\mathcal{F}_d$. Therefore this probability is the same for every f.

We prove now this fact by induction on $|S| + n$; it is trivial for $|S| = n = 0$. Suppose that, after removing the first bit of each element of S, we still get $|S|$ distinct vectors; then we can apply the induction hypothesis with S and $n - 1$. Otherwise, there are disjoint subsets $A, B, C \subseteq \{0,1\}^{n-1}$ such that $S = \{0,1\} \times A \cup \{0\} \times B \cup \{1\} \times C$, and $A \neq \emptyset$.

We can find, by induction, a polynomial $p_{0A,0B,1C}$ of degree $\leq d$ on $n - 1$ variables that computes f on $\{0\} \times A \cup \{0\} \times B \cup \{1\} \times C$. As $|S| = 2|A| + |B| + |C|$, either $|A| + |B|$ or $|A| + |C|$ is at most $\frac{|S|}{2} < 2^d$; assume the latter. Then any function $g : A \cup C \to \mathbb{F}$ can be evaluated by some polynomial $p_{AC}(y)$ of degree $\leq d - 1$; consider $g(y) = 0$ if $y \in C$ and $g(y) = f(1, y) - p_{0A,0B,1C}(1, y)$ if $y \in A$. Then the polynomial $p(x, y) = p_{0A,0B,1C}(y) + xp_{AC}(y)$ does the job. □

6.2.4 Proof of item 2 of Proposition 6.1. We show that there are $(q, 1/6)$-regular functions f that can be computed by small circuits. For this we need the following theorem:

Theorem 6.2. ([AS92]) *It is possible to construct B bits that are r-wise independent using $O(r \log B)$ random bits.*

Moreover, the construction can be carried out in $\mathcal{NC}$; that is, there is a bounded fan-in circuit of depth $O(\text{polylog}(r \log B))$ and polynomial size that, given as input $i \in [B]$ and m random bits, computes the i-th variable.

Putting $B \triangleq |\{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}|$, $r = n^4$, we see that the family of functions $f : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$ given by Theorem 6.2 is n^4-independent. Furthermore, each f is in $\mathcal{NC}$. Taking the closure of this family under G_n (considered as a multiset) we obtain a n^4-uniform family. By Lemma 6.2, there is a way to fix the $\text{poly}(n)$ random bits so that the resulting function is $(n - O(\log n), 1/6)$-regular.

7 Lower bounds for testing size-s Boolean circuits and degree-d Boolean functions

Theorem 7.1. *There is a constant $c > 0$ such that* for all $s \leq n^c$ *testing size-s Boolean circuits requires $\Omega(s^{1/c})$ queries.*

Proof. By Corollary 6.1, for all r there is a function $f' : \{0,1\}^r \to \{0,1\}$ such that f' can be computed by circuits of size r^c (for some constant c depending on the depth of the circuit computing f' that is guaranteed by Corollary 6.1) and if $g' : \{0,1\}^r \to \{0,1\}$ is a truncated random function then any algorithm that makes $o(r)$ queries cannot distinguish a random permutation of f' from a random permutation of $ext(g')$. Now with high probability the random truncated function g' will be far from all functions computed by circuits of size $2^{\Theta(n)} \gg r^c$. Hence we have functions $f', g' : \{0,1\}^r \to \{0,1\}$ such that f' can be computed by circuits of size r^c and g' is far from all functions computed by circuits of size $2^{\Theta(n)} \gg r^c$, yet any algorithm that makes $o(r)$ queries cannot distinguish a random permutation of f' from a random permutation of g'.

We can choose $r = \Theta(s^{1/c})$. Given f, g' and before, consider their padding extensions $f, g : \{0,1\}^n \to \{0,1\}$, where $f = pad(f')$ and $g = pad(g')$. define $f : \{0,1\}^n \to \{0,1\}$ to be the padding extension of f', where From Lemma 6.1 we obtain that any algorithm making $o(r)$ queries cannot distinguish a random permutation of f from a random permutation of g. Since the extension does not change the size of the Boolean circuit that computes the corresponding functions, the query complexity of testing a function of size-s Boolean circuits is $\Omega(r) = \Omega(s^{1/c})$. □

As any k junta can be written as a polynomial of degree[4] at most k, whereas almost all truncated functions are far from all polynomials of degree $n-\Theta(1)$, Theorem 6.1 implies the following:

Theorem 7.2. *The query complexity of testing whether a function $f : \{0,1\}^n \to \{0,1\}$ has degree at most d is $\Omega(d)$, for any $d \leq n - \omega(1)$.*

8 $O(k \log k)$ upper bound for testing isomorphism to k-juntas

Theorem 8.1. *Isomorphism to any k-junta can be tested with $O(\frac{k \log k}{\epsilon^2})$ queries.*

High-level overview of the proof. The first ingredient in our proof is a tolerant, noise-resistant and bias-resistant isomorphism tester RobustIsoTest (Algorithm 2 below). Informally, RobustIsoTest allows us to test isomorphism of an unknown g to a known function f, even if instead of an oracle access to g we are given a sampler that produces pairs (x, a), where

- there is some h that is close to g, and $\Pr[h(x) = a]$ is large;
- the distribution of the x's from the sampled pairs is close to uniform.

The basic idea that allows us to use RobustIsoTest for testing isomorphism to k-juntas is the following: if we could simulate a noisy almost-uniform sampler to the core of h, where $h : \{0,1\}^n \to \{0,1\}$ is the presumed k-junta that is close to $g : \{0,1\}^n \to \{0,1\}$, then we could test whether g is isomorphic to f. What we show is, roughly speaking, that for the aforementioned simulation it suffices to detect k disjoint subsets $J_1, \ldots, J_k \subseteq [n]$ such that each subset contains at most one relevant variable of the presumed k-junta $h : \{0,1\}^n \to \{0,1\}$.

To obtain such sets we use the second ingredient, which is the optimal junta tester of Blais [Bla09]. This tester, in addition to testing whether g is a k-junta, can provide (in case g is close to some k-junta h) a set of $\leq k$ blocks (sets of indices), such that each block contains exactly one of the relevant variables of h. The trouble is that the k-junta h may not be the closest one to g. In fact, even if g is a k-junta itself, h may be some other function that is only close to g. Taking these considerations into account constitutes the bulk of the proof.

8.1 Testing isomorphism between the cores.

In the following we use the term *black-box algorithm* for algorithms that take no input.

Definition 8.1. *Let $g : \{0,1\}^k \to \{0,1\}$ be a function, and let $\eta, \mu \in [0,1)$. An (η,μ)-*noisy sampler for g *is a black-box probabilistic algorithm $\widetilde{g}$ that on each execution outputs $(x, a) \in \{0,1\}^k \times \{0,1\}$ such that*

- *$x \in \{0,1\}^k$ is distributed according to some distribution $\mathcal{D}$ on $\{0,1\}^k$, such that the total variation distance between $\mathcal{D}$ and the uniform distribution is at most μ; namely, for all $A \subseteq \{0,1\}^k$, $\left|\Pr_{x\sim\mathcal{D}}[x \in A] - |A|/2^k\right| \leq \mu$;*
- *$\Pr[a = g(x)] \geq 1 - \eta$,*

where the probability is taken over the randomness of $\widetilde{g}$, which also determines x.

We stress that the two items are **not** necessarily independent; e.g., it may be that for some $\alpha \in \{0,1\}^k$, $\Pr[a = g(x) \mid x = \alpha] = 0$.

The following is essentially a strengthening of Occam's razor that is both tolerant, noise-resistant and bias-resistant:

Proposition 8.1. *There is an algorithm* RobustIsoTest *that, given $\epsilon \in \mathbb{R}^+$, $k \in \mathbb{N}$, a function $f : \{0,1\}^k \to \{0,1\}$ and a (η,μ)-noisy sampler $\widetilde{g}$*

[4] The term "degree" here refers to the degree of $f : \{0,1\}^n \to \{0,1\}$ when viewed as a polynomial $p \in \mathbb{F}[x_1, \ldots, x_n]$ with coefficients in some field $\mathbb{F}$. (In particular, when $\mathbb{F} = \mathbb{Q}$ we speak of the Fourier degree).

for some $g : \{0,1\}^k \to \{0,1\}$*, where* $\eta \leq \epsilon/100$ *and* $\mu \leq \epsilon/10$*, satisfies the following:*

- *if* $\mathrm{distiso}(f, g) < \epsilon/10$*, it accepts with probability at least* $9/10$*;*
- *if* $\mathrm{distiso}(f, g) > 9\epsilon/10$*, it rejects with probability at least* $9/10$*;*
- *it draws* $O(\frac{k \log k}{\epsilon^2})$ *samples from* $\widetilde{g}$*.*

Proof. Consider the tester described in Algorithm 2. It

Algorithm 2 (RobustIsoTest – tests if $f \cong g$, tolerantly with noise)

---

let $q \leftarrow \frac{c \log(k!)}{\epsilon^2}$, where c is a constant chosen later
obtain q independent samples $(x^1, a^1), \ldots, (x^q, a^q)$ from $\widetilde{g}$
accept if and only if there exists a permutation π of $[k]$ such that $\left|\left\{i \in [q] : f^\pi(x^i) \neq a^i\right\}\right| < \epsilon q/2$.

---

is clear that RobustIsoTest uses $O(\frac{k \log k}{\epsilon^2})$ queries.

Fix a permutation π. Let $\delta_\pi = \mathrm{dist}(f^\pi, g)$ and let $\Delta_\pi \subseteq \{0,1\}^k$, $|\Delta_\pi| = \delta_\pi 2^k$, be the set of inputs on which f^π and g disagree. Since the x's are independent random variables, distributed according to some distribution $\mathcal{D}$ that is μ-close to uniform, we have

$$\zeta_\pi \triangleq \Pr_{x \sim \mathcal{D}}[x \in \Delta_\pi] = \delta_\pi \pm \mu$$

(by $a = b \pm c$ we mean $|a - b| \leq c$).

Using Chernoff bounds (additive form) we can upper bound

$$\Pr\left[\left||\{i \in [q] : f^\pi(x^i) \neq g(x^i)\}| - \zeta_\pi q\right| > \epsilon q/10\right]$$

by $2^{-\Omega(\epsilon^2 q)}$, which is less than $\frac{1}{20(k!)}$ for sufficiently large constant c. Therefore, with probability at least $19/20$,

$$|\{i \in [q] : f^\pi(x^i) \neq g(x^i)\}| = \zeta_\pi q \pm \epsilon q/10 =$$
$$= \delta_\pi q \pm (\mu q + \epsilon q/10)$$

holds for all permutations π. To relate this to the fraction of samples (x, a) for which $f^\pi(x) \neq a$, we use Markov's inequality:

$$\Pr\left[|\{i \in [q] : a^i \neq g(x^i)\}| \geq \epsilon q/5\right] \leq$$
$$\leq \Pr\left[|\{i \in [q] : a^i \neq g(x^i)\}| \geq 20\eta q\right] \leq 1/20.$$

Therefore, with probability at least $9/10$,

$$\begin{aligned} |\{i \in [q] : f^\pi(x^i) \neq a^i\}| &= \delta_\pi q \pm (\mu q + 3\epsilon q/10) \\ &= \delta_\pi q \pm 2\epsilon q/5 \end{aligned}$$

for all π.

The result follows, since if $\mathrm{distiso}(f, g) < \epsilon/10$ then there exists π such that $\delta_\pi q + 2\epsilon q/5 < \epsilon q/2$; and if $\mathrm{distiso}(f, g) > 9\epsilon/10$ then for all π, $\delta_\pi q - 2\epsilon q/5 > \epsilon q/2$. □

8.2 Some definitions and lemmas.

Definition 8.2. *Given a* k*-junta* $f : \{0,1\}^n \to \{0,1\}$ *we define* $\mathsf{core}_k(f) : \{0,1\}^k \to \{0,1\}$ *to be the restriction of* f *to its relevant variables (where the variables are placed according to the natural order). In case* f *has less than* k *relevant variables,* $\mathsf{core}_k(f)$ *is extended to a* $\{0,1\}^k \to \{0,1\}$ *function by adding dummy variables.*

Throughout this section, a random partition $\mathcal{I} = I_1, \ldots, I_\ell$ of $[n]$ into ℓ sets is constructed by starting with ℓ empty sets, and then putting each coordinate $i \in [n]$ into one of the ℓ sets picked uniformly at random. Unless explicitly mentioned otherwise, $\mathcal{I}$ will always denote a random partition $\mathcal{I} = I_1, \ldots, I_\ell$ of $[n]$ into ℓ subsets, where ℓ is even; and $\mathcal{J} = J_1, \ldots, J_k$ will denote an (ordered) k-subset of $\mathcal{I}$ (meaning that there are $a_1, \ldots, a_k$ such that $J_i = I_{a_i}$ for all $i \in [k]$).

Definition 8.3. (Operators $\mathsf{replicate}$ and $\mathsf{extract}$) *We call* $y \in \{0,1\}^n$ $\mathcal{I}$-regular *if the restriction of* y *on every set of* $\mathcal{I}$ *is constant; that is, if for all* $i \in [\ell]$ *and* $j, j' \in I_i$*,* $y_j = y_{j'}$*.*

- *Given* $z \in \{0,1\}^\ell$*, define* $\mathsf{replicate}_\mathcal{I}(z)$ *to be the* $\mathcal{I}$*-regular string* $y \in \{0,1\}^n$ *obtained by setting* $y_j \leftarrow z_i$ *for all* $i \in \ell$ *and* $j \in I_i$*.*
- *Given an* $\mathcal{I}$*-regular* $y \in \{0,1\}^n$ *and an ordered subset* $\mathcal{J} = (J_1, \ldots, J_k)$ *of* $\mathcal{I}$ *define* $\mathsf{extract}_{\mathcal{I},\mathcal{J}}(y)$ *to be the string* $x \in \{0,1\}^k$ *where for every* $i \in [k]$*:* $x_i = y_j$ *if* $J_i \neq \emptyset$ *and* $j \in J_i$*; and* x_i *is a uniformly random bit if* $J_i = \emptyset$*.*

Definition 8.4. (Distributions $\mathcal{D}_\mathcal{I}$ and $\mathcal{D}_\mathcal{J}$) *For any* $\mathcal{I}$ *and* $\mathcal{J} \subseteq \mathcal{I}$ *as above, we define a pair of distributions:*

- *The distribution* $\mathcal{D}_\mathcal{I}$ *on* $\{0,1\}^n$*: A random* $y \sim \mathcal{D}_\mathcal{I}$ *is obtained by*
  1. *picking* $z \in \{0,1\}^\ell$ *uniformly at random among all* $\binom{\ell}{\ell/2}$ *strings of weight* $\ell/2$*;*
  2. *setting* $y \leftarrow \mathsf{replicate}_\mathcal{I}(z)$*.*
- *The distribution* $\mathcal{D}_\mathcal{J}$ *on* $\{0,1\}^{|\mathcal{J}|}$*: A random* $x \sim \mathcal{D}_\mathcal{J}$ *is obtained by*
  1. *picking* $y \in \{0,1\}^n$ *at random, according to* $\mathcal{D}_\mathcal{I}$*;*

2. setting $x \leftarrow \mathsf{extract}_{\mathcal{I},\mathcal{J}}(y)$.

LEMMA 8.1. (PROPERTIES OF $\mathcal{D}_{\mathcal{I}}$ AND $\mathcal{D}_{\mathcal{J}}$)

1. *For all* $\alpha \in \{0,1\}^n$, $\Pr_{\mathcal{I},y\sim\mathcal{D}_{\mathcal{I}}}[y=\alpha] = 1/2^n$*;*

2. *Assume* $\ell > 4k^2$*. For every* $\mathcal{I}$ *and* $\mathcal{J} \subseteq \mathcal{I}$*, the distance in the* L_∞ *norm between* $\mathcal{D}_{\mathcal{J}}$ *and the uniform distribution on* $\{0,1\}^{|\mathcal{J}|}$ *is bounded by* $2^{-k}4|\mathcal{J}|^2/\ell$*, and therefore the total variation distance between the two is at most* $4|\mathcal{J}|^2/\ell$*.*

Proof. 1. Each choice of $z \in \{0,1\}^\ell$, $|z| = \ell/2$, in Definition 8.4 splits $\mathcal{I}$ into two equally-sized sets: $\mathcal{I}^0$ and $\mathcal{I}^1$; and the bits corresponding to indices in $\mathcal{I}^b$ (where $b \in \{0,1\}$) are set to b in the construction of y. For each index $i \in [n]$, the block it is assigned to is chosen independently at random from $\mathcal{I}$, and therefore falls within $\mathcal{I}^0$ (or $\mathcal{I}^1$) with probability $1/2$, independently of other $j \in [n]$. (This actually shows that the first item of the lemma still holds if z is an arbitrarily fixed string of weight $\ell/2$, rather than a randomly chosen one).

2. Let $k = |\mathcal{J}|$. We only need to take care of the case were all sets J_i in $\mathcal{J}$ are non-empty; having empty sets can only decrease the distance to uniform. Let $w \in \{0,1\}^k$. The choice of $y \sim \mathcal{D}_{\mathcal{I}}$, in the process of obtaining $x \sim \mathcal{D}_{\mathcal{J}}$, is independent of $\mathcal{J}$; thus, for every $i \in [k]$ we have

$$\Pr_{x\sim\mathcal{D}_{\mathcal{J}}}[x_i = w_i \mid x_j = w_j \ \forall j < i] \le \frac{\ell/2}{\ell-k} < \frac{1}{2} + \frac{k}{\ell},$$

and

$$\Pr_{x\sim\mathcal{D}_{\mathcal{J}}}[x_i = w_i \mid x_j = w_j \ \forall j < i] \ge \frac{\ell/2-k}{\ell-k} > \frac{1}{2} - \frac{k}{\ell}.$$

Using the inequalities $1 - my \le (1-y)^m$ for all $y < 1, m \in \mathbb{N}$ and $(1+y)^m \le e^{my} \le 1 + 2my$ for all $m \ge 0, 0 \le my \le 1/2$, we conclude

$$\Pr_{x\sim\mathcal{D}_{\mathcal{J}}}[x = w] = \left(\frac{1}{2} \pm \frac{k}{\ell}\right)^k = \frac{1}{2^k}\left(1 \pm \frac{4k^2}{\ell}\right).$$

whereas a truly uniform distribution U should satisfy $\Pr_{x\sim U}[x = w] = 1/2^k$. Hence the total variation distance between U and $\mathcal{D}_{\mathcal{J}}$ is at most $4k^2/\ell$.

□

DEFINITION 8.5. (BLACK-BOX ALGORITHM $\mathsf{sampler}$) *Given* $\mathcal{I}, \mathcal{J}$ *as above and oracle access to* $g : \{0,1\}^n \to \{0,1\}$*, we define a probabilistic black-box algorithm* $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ *that on each execution produces a pair* $(x,a) \in \{0,1\}^{|\mathcal{J}|} \times \{0,1\}$ *as follows: it picks a random* $y \sim \mathcal{D}_{\mathcal{I}}$ *and outputs the pair* $(\mathsf{extract}_{\mathcal{I},\mathcal{J}}(y), g(y))$*.*

Note that just one query is made to g in every execution of $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$. Notice also that the x in the pairs $(x,a) \in \{0,1\}^{|\mathcal{J}|} \times \{0,1\}$ produced by $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is distributed according to distribution $\mathcal{D}_{\mathcal{J}}$ defined above.

8.3 From junta-testers to noisy-samplers. Throughout this section Jun_k will denote the class of k-juntas (on n variables), and for $A \subseteq [n]$, Jun_A will denote the class of juntas with all relevant variables in A. In addition, given a function $g : \{0,1\}^n \to \{0,1\}$, we denote by $g^* : \{0,1\}^n \to \{0,1\}$ the k-junta that is closest to g (if there are several k-juntas that are equally close, break ties using some arbitrarily fixed scheme). Clearly, if g is itself a k-junta then $g^* = g$.

LEMMA 8.2. *[FKR+02] For any* $f : \{0,1\}^n \to \{0,1\}$ *and* $A \subseteq [n]$

$$\mathrm{dist}(f, \mathsf{Jun}_A) \le \mathit{Inf}_f([n] \setminus A) \le 2 \cdot \mathrm{dist}(f, \mathsf{Jun}_A).$$

We will also use the fact (see [FKR+02, Bla09] for a proof) that influence is monotone and subadditive; namely, for all $f : \{0,1\}^n \to \{0,1\}$ and $A, B \subseteq [n]$,

$$\mathit{Inf}_f(A) \le \mathit{Inf}_f(A \cup B) \le \mathit{Inf}_f(A) + \mathit{Inf}_f(B).$$

For the following definition and lemma we recall the distributions $\mathcal{D}_{\mathcal{I}}$ and $\mathcal{D}_{\mathcal{J}}$ from Definition 8.4.

DEFINITION 8.6. *Given* $\delta > 0$*, function* $g : \{0,1\}^n \to \{0,1\}$*, partition* $\mathcal{I} = I_1, \ldots, I_\ell$ *of* $[n]$ *and a* k*-subset* $\mathcal{J}$ *of* $\mathcal{I}$ *(where* $\ell > 4k^2$*), we call the pair* $(\mathcal{I}, \mathcal{J})$ δ-good *(with respect to* g*) if there exists a* k*-junta* $h : \{0,1\}^n \to \{0,1\}$ *such that the following conditions are satisfied.*

1. *Conditions on* h*:*
   (a) *Every relevant variable of* h *is also a relevant variable of* g^* *(recall that* g^* *denotes the* k*-junta closest to* g*);*
   (b) $\mathrm{dist}(g^*, h) < \delta$.

2. *Conditions on* $\mathcal{I}$*:*
   (a) *For all* $j \in [\ell]$*,* I_j *contains at most one variable of* $\mathsf{core}_k(g^*)$*;* [5]

[5] Note that this, along with 1a, implies that every block I_j contains at most one relevant variable of h, since the variables of $\mathsf{core}_k(g^*)$ contain all relevant variables of g^*.

(b) $\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g(y) \neq g^*(y)] \leq 10 \cdot \mathrm{dist}(g, g^*)$*;*

3. *Conditions on* $\mathcal{J}$*:*

   (a) The set $\bigcup_{I_j\in\mathcal{J}} I_j$ *contains* all *relevant variables of* h*;*

Lemma 8.3. *Let* $\delta, g, \mathcal{I}$ *be as in the preceding definition. If the pair* $(\mathcal{I}, \mathcal{J})$ *is* δ*-good, then for some permutation* $\pi : [k] \to [k]$,

$$\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g(y) \neq \mathsf{core}_k(g^*)^\pi(\mathsf{extract}_{\mathcal{I},\mathcal{J}}(y))] <$$

$$< 2\delta + 8k^2/\ell + 10 \cdot \mathrm{dist}(g, g^*).$$

Proof. By item 2b in Definition 8.6, it suffices to prove that

$$\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g^*(y) \neq \mathsf{core}_k(g^*)^\pi(\mathsf{extract}_{\mathcal{I},\mathcal{J}}(y))] < 2\delta + 8k^2/\ell$$

for some π.

Let h be the k-junta that witnesses the fact that the pair $(\mathcal{I}, \mathcal{J})$ is δ-good. Let $V \subseteq [n]$ be the set of k variables of $\mathsf{core}_k(g^*)$. (Recall that V may actually be a superset of the relevant variables of g^*.) Let $\mathcal{J}' \triangleq \{I_j \in \mathcal{I} : I_j \cap V \neq \emptyset\}$ be an ordered subset respecting the order of $\mathcal{J}$, and let π be the permutation that maps the i-th relevant variable of g^* (in the standard order) to the index $\pi(i)$ of the element of $\mathcal{J}'$ in which it is contained. We assume without loss of generality that π is the identity map.

It follows from Definition 8.6 that $|\mathcal{J}'| = |V| = k$, since each block in $\mathcal{I}$ contains at most one variable of $\mathsf{core}_k(g^*)$. For any $\mathcal{I}$-uniform $y \in \{0,1\}^n$, let $x \triangleq \mathsf{extract}_{I,\mathcal{J}}(y)$ and $x' \triangleq \mathsf{extract}_{I,\mathcal{J}'}(y)$ denote the k-bit strings corresponding to $\mathcal{J}$ and $\mathcal{J}'$. By definitions, we have the equalities

(1) $\quad g^*(y) = \mathsf{core}_k(g^*)(x')$,

(2) $\quad \mathsf{core}_k(h)(x) = \mathsf{core}_k(h)(x')$.

The first equality is by Definition 8.3, and the second one follows from items 1a and 3a in Definition 8.6. From item 1b we also have

(3) $\quad \Pr_{r\in\{0,1\}^k}[\mathsf{core}_k(g^*)(r) \neq \mathsf{core}_k(h)(r)] < \delta$,

where r is picked uniformly at random. However, by the second item of Lemma 8.1, the distribution $\mathcal{D}_\mathcal{J}$ is $4k^2/\ell$ close to uniform [6]; combining this with (3) we also get

(4) $\quad \Pr_{y\sim\mathcal{D}_\mathcal{I}}[\mathsf{core}_k(g^*)(x) \neq \mathsf{core}_k(h)(x)] < \delta + 4k^2/\ell$.

Likewise, we have

(5) $\quad \Pr_{y\sim\mathcal{D}_\mathcal{I}}[\mathsf{core}_k(g^*)(x') \neq \mathsf{core}_k(h)(x')] < \delta + 4k^2/\ell$,

[6] Recall that $\mathcal{D}_\mathcal{J}$ is a distribution on $\{0,1\}^k$, where a random $x \sim \mathcal{D}_\mathcal{J}$ is obtained by picking a random $y \sim \mathcal{D}_\mathcal{I}$ and setting $x \leftarrow \mathsf{extract}_{\mathcal{I},\mathcal{J}}(y)$.

thus, using $(2, 4, 5)$ and the union bound we get

(6) $\quad \Pr_{y\sim\mathcal{D}_\mathcal{I}}[\mathsf{core}_k(g^*)(x') \neq \mathsf{core}_k(g^*)(x)] < 2\delta + 8k^2/\ell$.

Combining (1) and (6) we conclude that

$$\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g^*(y) \neq \mathsf{core}_k(g^*)(x)] < 2\delta + 8k^2/\ell,$$

and the claim follows. ☐

Corollary 8.1. *If the pair* $(\mathcal{I}, \mathcal{J})$ *is* δ*-good (with respect to* g*), then* $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ *is* (η, μ)*-noisy sampler for a permutation of* $\mathsf{core}_k(g^*)$*, with* $\eta \leq 2\delta + 8k^2/\ell + 10 \cdot \mathrm{dist}(g, g^*)$ *and* $\mu \leq 4k^2/\ell$*.*

Proof. Recall that $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is a probabilistic black-box algorithm, that on each execution produces a pair $(x, a) \in \{0,1\}^k \times \{0,1\}$ as follows: it picks a random $y \sim \mathcal{D}_\mathcal{I}$ and outputs the pair $(x, a) \triangleq (\mathsf{extract}_{\mathcal{I},\mathcal{J}}(y), g(y))$.

To be an (η, μ)-noisy sampler for $\mathsf{core}_k(g^*)^\pi$, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ has to satisfy the following:

- the distribution of $x \in \{0,1\}^k$ in its pairs should be μ close to uniform (in total variation distance);
- $\Pr_{(x,a)\leftarrow\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)}\left[a = \mathsf{core}_k(g^*)^\pi(x)\right] \geq 1 - \eta$.

The first item follows from the second item of Lemma 8.1. The second item follows from Lemma 8.3. ☐

Now we set up a version of the junta tester from [Bla09] that is needed for our algorithm. A careful examination of the proof in [Bla09] yields the following:

Theorem 8.2. (Corollary to [Bla09]) *The property* Jun_k *can be tested with one-sided error using* $O(k \log k + k/\epsilon)$ *queries.*

Moreover, the tester $\mathrm{T}_{[Bla09]}$ *can take a (random) partition* $\mathcal{I} = I_1, \ldots, I_\ell$ *of* $[n]$ *as input, where* $\ell = \ell_{[Bla09]}(k, \epsilon) = \Theta(k^9/\epsilon^5)$ *is even, and output (in case of acceptance) a* k*-subset* $\mathcal{J}$ *of* $\mathcal{I}$ *such that for any* g *the following conditions hold (the probabilities below are taken over the randomness of the tester and the construction of* $\mathcal{I}$*):*

- *if* g *is a* k*-junta,* $\mathrm{T}_{[Bla09]}$ *always accepts;*
- *if* g *is* $\epsilon/2400$*-far from* Jun_k*, then* $\mathrm{T}_{[Bla09]}$ *rejects with probability at least* $9/10$*;*
- *for* any g*, with probability at least* $4/5$ *either* $\mathrm{T}_{[Bla09]}$ *rejects, or it outputs* $\mathcal{J}$ *such that the pair* $(\mathcal{I}, \mathcal{J})$ *is* $\epsilon/600$*-good (as per Definition 8.6). (In particular, if* g *is a* k*-junta then with probability at least* $4/5$*,* $\mathrm{T}_{[Bla09]}$ *outputs a set* $\mathcal{J}$ *such that* $(\mathcal{I}, \mathcal{J})$ *is* $\epsilon/600$*-good.)*

Proof. In view of the results stated in [Bla09], only the last item needs justification. [7]

We start with a brief description of how $T_{[Bla09]}$ works. Given the partition $\mathcal{I}$, $T_{[Bla09]}$ starts with an empty set $S = \emptyset$, and iteratively finds indices $j \in [\ell] \setminus S$ such that for some pair of inputs $y, y' \in \{0,1\}^n$, $y_{\upharpoonright_{[n]\setminus I_j}} = y'_{\upharpoonright_{[n]\setminus I_j}}$ but $g(y) \neq g(y')$. In other words, it finds j such that I_j contains at least one influential variable (let us call such a block I_j *relevant*). Then j is joined to S, and the algorithm proceeds to the next iteration. $T_{[Bla09]}$ stops at some stage, and rejects if and only if $|S| > k$. If g is not rejected (i.e. if $T_{[Bla09]}$ terminates with $|S| \leq k$), then

$(*)$ with probability at least $19/20$ the set S satisfies $Inf_g\Big([n] \setminus (\bigcup_{j\in S} I_j)\Big) \leq \epsilon/4800$.

We will use this S to construct the subset $\mathcal{J} \subseteq \mathcal{I}$ as follows:

- for every $j \in S$, we put the block I_j into $\mathcal{J}$;
- if $|S| < k$ then we extend $\mathcal{J}$ by putting in it $k - |S|$ additional "dummy" blocks from $\mathcal{I}$ (some of them possibly empty), obtaining a set $\mathcal{J}$ of size exactly k.

Now we go back to proving the third item of Theorem 8.2. Recall that g^* denotes the closest k-junta to g. Let $R \in \binom{[n]}{\leq k}$ denote the set of the relevant variables of g^*, and let $V \in \binom{[n]}{k}$, $V \supseteq R$, denote the set of the variables of $\mathsf{core}_k(g^*)$. Assume that $\mathrm{dist}(g, \mathsf{Jun}_k) \leq \epsilon/2400$, [8] and $T_{[Bla09]}$ did not reject. In this case,

- by $(*)$, with probability at least $19/20$ the set $\mathcal{J}$ satisfies
$$\begin{aligned} Inf_g\Big([n] \setminus (\bigcup_{I_j\in\mathcal{J}} I_j)\Big) &\leq Inf_g\Big([n] \setminus (\bigcup_{j\in S} I_j)\Big) \\ &\leq \epsilon/4800; \end{aligned}$$
- since $\ell \gg k^2$, with probability larger than $19/20$ all elements of V fall into different blocks of the partition $\mathcal{I}$;
- by Lemma 8.1, $\Pr_{\mathcal{I}, y\sim\mathcal{D}_\mathcal{I}}\Big[g(y) = g^*(y)\Big] = \mathrm{dist}(g, g^*)$; hence by Markov's inequality, with probability at least $9/10$ the partition $\mathcal{I}$ satisfies $\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g(y) \neq g^*(y)] \leq 10 \cdot \mathrm{dist}(g, g^*)$.

[7]The somewhat different constants can be easily achieved by increasing (by a constant factor) the number of iterations and partition sizes of the algorithm.

[8]For other g's the third item follows from the second item.

So with probability at least $4/5$, all three of these events occur. Now we show that conditioned on them, the pair $(\mathcal{I}, \mathcal{J})$ is $\epsilon/600$-good.

Let $U = R \cap (\bigcup_{I_j\in\mathcal{J}} I_j)$. Informally, U is the subset of the relevant variables of g^* that were successfully "discovered" by $T_{[Bla09]}$. Since $\mathrm{dist}(g, g^*) \leq \epsilon/2400$, we have $Inf_g([n] \setminus V) \leq \epsilon/1200$ (by Lemma 8.2). By the subadditivity and monotonicity of influence we get

$$\begin{aligned} &Inf_g([n] \setminus U) \leq Inf_g([n] \setminus V) + Inf_g(V \setminus U) \leq \\ &\leq Inf_g([n] \setminus V) + Inf_g\Big([n] \setminus (\bigcup_{I_j\in\mathcal{J}} I_j)\Big) \leq \epsilon/960, \end{aligned}$$

where the second inequality follows from $V \setminus U \subseteq [n] \setminus (\bigcup_{I_j\in\mathcal{J}} I_j)$. This means, by Lemma 8.2, that there is a k-junta h in Jun_U satisfying $\mathrm{dist}(g, h) \leq \epsilon/960$, and by triangle inequality, $\mathrm{dist}(g^*, h) \leq \epsilon/2400 + \epsilon/960 < \epsilon/600$. Based on this h, we can verify that the pair $(\mathcal{I}, \mathcal{J})$ is $\epsilon/600$-good by going over the conditions in Definition 8.6. □

8.4 Putting everything together.

Consider the tester described in Algorithm 3. The proof of Theorem 8.1 follows from the next lemma:

Algorithm 3 (tests isomorphism to a k-junta f)

1: let $\ell = \ell_{[Bla09]}(k, \epsilon) = \Theta(k^9/\epsilon^5)$
2: randomly partition $[n]$ into $\mathcal{I} = (I_1, \ldots, I_\ell)$
3: test g for being a k-junta, using $T_{[Bla09]}$ with $\mathcal{I} = I_1, \ldots, I_\ell$ (see Theorem 8.2)
4: **if** $T_{[Bla09]}$ rejects **then**
5: reject
6: **end if**
7: let $\mathcal{J} \subseteq \mathcal{I}$ be the set output by $T_{[Bla09]}$
8: construct $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ (see Section 8.2)
9: accept iff RobustIsoTest($\mathsf{core}_k(f), \mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$) accepts (see Section 8.1)

Lemma 8.4. *Algorithm 3 satisfies the following conditions:*

- *if* $g \cong f$ *then it accepts with probability at least* $2/3$*;*
- *if* $\mathrm{distiso}(f, g) \geq \epsilon$ *then it rejects with probability at least* $2/3$*;*
- *its query complexity is* $O(k \log k/\epsilon^2)$.

Proof of item 1. Assume $g \cong f$, and hence $\mathsf{core}_k(g) \cong \mathsf{core}_k(f)$. Since g is a k-junta, Algorithm 3 does not reject on line 5, because $T_{[Bla09]}$ has one-sided error. So in this case, by Theorem 8.2, with probability at least $4/5$ the pair $(\mathcal{I}, \mathcal{J})$ is $\epsilon/600$-good. If so, by Corollary 8.1, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is a (η,μ)-noisy sampler for a function isomorphic to $\mathsf{core}_k(g^*) = \mathsf{core}_k(g)$, where $\eta \leq 2\epsilon/600 + 8k^2/\ell + 10 \cdot 0 < \epsilon/100$ and $\mu \leq 4k^2/\ell < \epsilon/10$, and hence RobustIsoTest accepts with probability at least $9/10$. Thus the overall acceptance probability is at least $2/3$.

Proof of item 2. If $\mathrm{distiso}(f,g) \geq \epsilon$ then one of the following must hold:

- either g is $\epsilon/2400$-far from Jun_k,
- or $\mathrm{dist}(g, \mathsf{Jun}_k) = \mathrm{dist}(g, g^*) \leq \epsilon/2400$ and $\mathrm{distiso}(\mathsf{core}_k(f), \mathsf{core}_k(g^*)) \geq \epsilon - \epsilon/2400 > 9\epsilon/10$.

If the first case holds, then $\mathrm{T}_{[\mathrm{Bla09}]}$ rejects with probability greater than $2/3$ and we are done. So assume that the second case holds.

By the third item of Theorem 8.2, with probability at least $4/5$, $\mathrm{T}_{[\mathrm{Bla09}]}$ either rejects g, or the pair $(\mathcal{I},\mathcal{J})$ is $\epsilon/600$ good. If $\mathrm{T}_{[\mathrm{Bla09}]}$ rejects then we are done. Otherwise, if an $\epsilon/600$-good pair is obtained, then by Corollary 8.1, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is a (η,μ)-noisy sampler for a function isomorphic to $\mathsf{core}_k(g^*)$, where $\eta \leq 2\epsilon/600 + 8k^2/\ell + 10 \cdot \epsilon/2400 < \epsilon/100$ and $\mu \leq 4k^2/\ell < \epsilon/10$, and hence RobustIsoTest rejects with probability at least $9/10$. Thus the overall rejection probability is at least $2/3$.

Proof of item 3. As for the query complexity, it is the sum of $O(k \log k + k/\epsilon)$ queries made by $\mathrm{T}_{[\mathrm{Bla09}]}$, and additional $O(k \log k/\epsilon^2)$ queries made by RobustIsoTest.

This completes the proof of Theorem 8.1. □

8.5 Query-efficient procedure for drawing random samples from the core. We conclude this section by observing that the tools developed above can be used for drawing random samples from the core of a k-junta g, so that generating each sample requires only one query to g.

PROPOSITION 8.2. *Let $\gamma > 0$ be an arbitrary constant. There is a randomized algorithm A, that given oracle access to any k-junta $g : \{0,1\}^n \to \{0,1\}$ satisfies:*

- *Algorithm A has two parts: preprocessor A_P and sampler A_S. A_P is executed only once; it makes $O(k \log k)$ queries to g, and produces a state $\alpha \in \{0,1\}^{\mathrm{poly}(n)}$. The sampler A_S can then be called on demand, with the state α as an argument; in each call, A_S makes only one query to g and outputs a pair $(x,a) \in \{0,1\}^k \times \{0,1\}$.*
- *With probability at least $4/5$, the state α produced by A_P is such that for some permutation $\pi : [k] \to [k]$,*

$$\Pr_{(x,a) \leftarrow A_S(\alpha)}[\mathsf{core}(g)^\pi(x) = \alpha] \geq 1 - \gamma.$$

Furthermore, the x's generated by the sampler A_S are independent random variables, distributed uniformly on $\{0,1\}^k$.

Proof. The preprocessor A_P starts by constructing a random partition $\mathcal{I}$ and calling the junta tester $\mathrm{T}_{[\mathrm{Bla09}]}$ with $\epsilon \triangleq \gamma$. Then A_P encodes in the state α the partition $\mathcal{I}$ and the subset $\mathcal{J} \subseteq \mathcal{I}$ output by $\mathrm{T}_{[\mathrm{Bla09}]}$ (see Theorem 8.2).

The sampler, given $\alpha = (\mathcal{I},\mathcal{J})$, obtains a pair $(x,a) \in \{0,1\}^k \times \{0,1\}$ by executing $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ (once). Then, with probability p_x (defined bellow), A_P outputs (x,a); and with probability $1-p_x$ it draws a uniformly random $z \in \{0,1\}^k$ and outputs $(z,0)$.

By Theorem 8.2 (third item), since g is a k-junta, with probability at least $4/5$, the pair $\mathcal{I},\mathcal{J}$ is $\epsilon/600$-good. So, by Corollary 8.1, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is a (η,μ)-noisy sampler for a function isomorphic to $\mathsf{core}_k(g^*) = \mathsf{core}_k(g)$, where $\eta \leq 2\epsilon/600 + 8k^2/\ell + 10 \cdot 0 < \epsilon/100$ and $\mu \leq 4k^2/\ell < \epsilon/100$. Moreover, the distribution of x in the pairs produced by $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ is $2^{-k}\mu < \epsilon 2^{-k}/100$ close to uniform in L_∞ norm. Since we need this distribution to be uniform, we use rejection sampling, with the only difference being that since $\mu \leq \epsilon/100 \ll 1$, we can stop after one execution of $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ at the cost of a small increase in the error probability.

Concretely, after drawing sample (x,a) from $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$, we accept it with probability

$$p_x \triangleq \frac{\Pr_{x_1 \sim U}[x_1 = x]}{(1+\mu)\Pr_{x_2 \sim D_{\mathcal{J}}}[x_2 = x]};$$

and with probability $1 - p_x$ we reject the sample (and output a uniformly random pair $(z,0)$ instead). It is easy to verify that the overall acceptance probability is $\mathbb{E}_{x \sim D_{\mathcal{J}}} p_x = 1/(1+\mu)$ and thus, conditioned on acceptance, the distribution of x is uniform. In the case of rejection (which occurs with probability $\mu/(1+\mu)$) it is uniform by definition; hence the overall distribution of x is uniform too, and $\Pr[a \neq g(x)] \leq \epsilon/100 + \mu/(1+\mu) < \epsilon/50 < \gamma$. □

9 Testing isomorphism between two unknown functions

An ϵ-tester for function isomorphism in the unknown-unknown setting is a probabilistic algorithm $\mathcal{A}$ that, given oracle access to two functions $f,g : \{0,1\}^n \to \{0,1\}$, satisfies the the following conditions: (1) if $f \cong g$ it accepts with probability at least $2/3$; (2) if $\mathrm{distiso}(f,g) \geq \epsilon$ it rejects with probability at least $2/3$. The query complexity of $\mathcal{A}$ is the worst-case number of queries it makes to f and g before making a decision. $\mathcal{A}$

is *non-adaptive* if its choice of queries does not depend on the outcomes of earlier queries. $\mathcal{A}$ has *one-sided error* if it always accepts in case $f \cong g$.

In the rest of the section we prove the following Theorem

PROPOSITION 9.1. *The following holds for any fixed $\epsilon > 0$.*

1. *There exists a non-adaptive one-sided ϵ-tester for function isomorphism in the unknown-unknown setting that has query complexity $O(2^{n/2}\sqrt{n \log n})$.*
2. *Any adaptive tester for function isomorphism in the unknown-unknown setting must have query complexity $\Omega(2^{n/2})$.*

9.1 Proof of Proposition 9.1, part 1: upper bound. In this section we show that isomorphism of a pair of unknown functions can be tested with a one-sided error non-adaptive tester that makes $O(2^{n/2}\sqrt{n \log n})$ queries. The tester is described in Algorithm 4.

Algorithm 4 (non-adaptive one-sided error tester for the unknown-unknown setting)

$Q \leftarrow \emptyset$
add every $x \in \{0,1\}^n$ to Q with probability $\sqrt{\frac{n \log n}{\epsilon 2^n}}$, independently of each other
if $|Q| > 10\sqrt{\frac{2^n}{\epsilon} n \log n}$ **then**
  accept
end if
query both f and g on all points in Q
accept if and only if there exists π such that for all $x \in Q$, either $f(x) = g(\pi(x))$ or $\pi(x) \notin Q$

It is clear that Algorithm 4 is non-adaptive, has one-sided error and it makes $O(2^{n/2}\sqrt{n \log n})$ queries. So we only need to prove that ϵ-far functions are accepted with probability at most 1/3. Since the event $|Q| \leq 10\sqrt{\frac{2^n}{\epsilon} n \log n}$ occurs with probability $1 - o(1)$, we can condition the rest of the argument over it. Let f and g be ϵ-far. That is, for all π there exist $\epsilon 2^n$ inputs $x \in \{0,1\}^n$ such that $f(x) \neq g(\pi(x))$. Fixed π and such an x, the probability that both x and $\pi(x)$ are in Q is at least $\frac{n \log n}{\epsilon 2^n}$, so any such π passes the acceptance condition with probability at most $(1 - n \log n / (\epsilon 2^n))^{\epsilon 2^n} \leq e^{-n \log n} = n^{-n} = o(1/n!)$. The proof follows by taking the union bound over all $n!$ permutations.

9.2 Proof of Proposition 9.1, part 2: lower bound. In this section we prove that any two-sided adaptive tester in the unknown-unknown setting must make $\Omega(2^{n/2})$ queries.

We define two distributions D_Y and D_N on pairs of functions such that any pair of functions drawn according to distribution D_Y are isomorphic, while any pair drawn according to distribution D_N is 1/8-far from isomorphic.

Recall from Definition 6.1 that a random truncated function $f : \{0,1\}^n \to \{0,1\}$ is defined as follows: if $\frac{n}{2} - \sqrt{n} \leq |x| \leq \frac{n}{2} + \sqrt{n}$ then $f(x) = 1$ with probability 1/2 and if $|x|$ is less than $\frac{n}{2} - \sqrt{n}$ or greater than $\frac{n}{2} + \sqrt{n}$ then $f(x) = 0$.

The distribution D_Y is constructed by letting the pair of functions consist of a random truncated function $f : \{0,1\}^n \to \{0,1\}$ and a function g that is obtained by permuting f using a random permutation in S_n.

For the distribution D_N the pair of functions are two independently chosen random truncated functions f and g. Now with probability $1 - o(1)$ the two functions are 1/8 far from each other (Observation 6.1). For any set $Q = \{x_1, \ldots, x_t\} \subseteq \{0,1\}^n$ of queries and any $p, q \in \{0,1\}^t$ let $\Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q)]$ be the probability that for all $1 \leq i \leq t$, $f(x_i) = p_i$ and $g(x_i) = q_i$ when f and g are drawn according to D_Y. Similarly we define $\Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)]$.

Without loss of generality we assume that $|x^i| \in [n/2 - \sqrt{n}, n/2 + \sqrt{n}]$ for all $i \in [t]$. By definition, if the pair f, g is drawn from D_N, the answers to the queries will be uniformly distributed, meaning that for any $p, q \in \{0,1\}^t$, we have

$$\Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)] = 1/2^{2t}.$$

Now let the pair be drawn according to D_Y and let π be the permutation on $[n]$ that defined the pair. Let E_Q denote the event that $\pi(Q)$ and Q are disjoint, that is, for all i, j inequality $\pi(x^i) \neq x^j$ holds. Conditioned on E_Q, the answers to the queries will again be distributed uniformly, that is

$$\Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q) | E_Q] = \Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)]$$

(note that the event in question is independent of E_Q when the pairs is drawn from D_N).

CLAIM 9.1. $\Pr[E_Q] \geq 2/3$.

Proof. [Of Claim 9.1] For any i and taking a random permutation π, the probability that $\pi(x^i) = x^j$ for some j is less than $t/\binom{n}{k}$ where $k = |x^i|$. Since $\frac{n}{2} - \sqrt{n} < k < \frac{n}{2} + \sqrt{n}$, this probability is bounded by $25t/2^n$. Hence, by the union bound, with probability $1 - \frac{25t^2}{2^n}$ for all i, j we have $\pi(x^i) \neq x^j$. So if $t < 2^{n/2}/10$, with probability at least 2/3 event E_Q happens. $\square$

Now $\Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q)]$ is at least

$$(\Pr[E_Q]) \Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q) \mid E_Q] \geq$$

$$\geq (2/3) \Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)].$$

This implies (by Lemma A.1) a lower bound of $2^{n/2}/10$ on the adaptive query complexity for the two-sided testing for the unknown-unknown setting.

10 Distinguishing two random functions with $\widetilde{O}(\sqrt{n})$ queries

In light of the fact that two trimmed random functions are hard to distinguish with fewer than roughly n queries, we may ask whether the restriction to trimmed functions is necessary. In this section we show that without such a restriction, the aforementioned task can be completed with only $\widetilde{O}(\sqrt{n})$ queries. We prove the following proposition, which says in particular that any function can be distinguished from a completely random function using $\widetilde{O}(\sqrt{n})$ queries.

PROPOSITION 10.1. *Let $\delta > 0$ be an arbitrary constant. For* any *function f and* any *distribution $\mathcal{D}_y$ over functions isomorphic to f, it is possible to distinguish $g \in \mathcal{D}_y$ from $g \in U$ with probability $1 - \delta$ using $\widetilde{O}(\sqrt{n})$ queries.*

Note that querying g only on inputs of Hamming weights $1, 2, n-1, n$ cannot help much. By querying the all-zero and all-one inputs, we can distinguish between the two cases only with probability $3/4$.[9] When considering singletons (and likewise, inputs of weight $n-1$), then f, g are isomorphic only if $|\{x \in L_1 : f(x) = 1\}| = |\{x \in L_1 : g(x) = 1\}|$. So a natural (and only) approach would be to test the equality of these measures by sampling. But notice that for most f, with very high probability (over the choice of g), these two measures will be at most $O(\sqrt{n})$ away from each other, which means that distinguishing the two cases requires at least $\Omega(n)$ samples.

Due to space constraints, the proof of Proposition is not included here. Here we present a sketch.

We show that $\widetilde{O}(\sqrt{n})$ queries into inputs of weight ≤ 2 are sufficient for distinguishing $g \in \mathcal{D}_y$ from $g \in U$ with high probability. One way to do this is to interpret the restriction of f and g to $\binom{[n]}{2}$ as adjacency functions of graphs on n vertices. It is not hard to prove that for any f and a randomly chosen g, the corresponding graphs G_f, G_g are 1/3-far from being isomorphic with overwhelming probability. On the other hand, if f is isomorphic to g then G_f is obviously isomorphic to G_g. Hence, we can use the isomorphism tester of [FM08] (in the appropriate setting) to distinguish between the two cases.

But in fact, the graph case is more complicated, since it is concerned with the worst case scenario (i.e., it should work for any pair of graphs). In our case, we only wish to distinguish a (possibly random) permutation of some given f from a random function g. Indeed, it turns out that we can reduce our problem directly to the task of testing equivalence of a samplable distribution to an explicitly given one. Then we can use an algorithm of Batu et al. [BFF+01] that solves exactly this problem with $\widetilde{O}(\sqrt{n})$ queries.

The formal details are worked out in the full version of the paper, available at `http://homepages.cwi.nl/~david/downloads/fiso.pdf`.

Acknowledgements. We thank Ronald de Wolf for many valuable comments. We also thank Noga Alon and Eric Blais for useful discussions.

[9] Notice that this success probability cannot be amplified, since the probability is taken over the choice of functions, rather than the randomness of the tester.

References

[AB10] Noga Alon and Eric Blais. Testing boolean function isomorphism. In *Proc. RANDOM-APPROX*, pages 394–405, 2010.

[AKK+03] Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, and Dana Ron. Testing low-degree polynomials over GF(2). In *Proc. RANDOM-APPROX*, pages 188–199, 2003.

[AS92] Noga Alon and Joel H. Spencer. *The Probabilistic Method.* Wiley, New York, 1992.

[BC10] Laszlo Babai and Sourav Chakraborty. Property testing of equivalence under a permutation group action. *To appear in The ACM Transactions on Computation Theory (ToCT)*, 2010.

[BEHL09] Ido Ben-Eliezer, Rani Hod, and Shachar Lovett. Random low degree polynomials are hard to approximate. In *Proc. RANDOM-APPROX*, pages 366–377, 2009.

[BFF+01] T. Batu, L. Fortnow, E. Fischer, R. Kumar, R. Rubinfeld, and P. White. Testing random variables for independence and identity. *Proc. IEEE Symposium on Foundations of Computer Science*, 0:442, 2001.

[Bla09] Eric Blais. Testing juntas nearly optimally. In *Proc. ACM symposium on the Theory of computing*, pages 151–158, New York, NY, USA, 2009. ACM.

[BO10] Eric Blais and Ryan O'Donnell. Lower bounds for testing function isomorphism. In *IEEE Conference on Computational Complexity*, pages 235–246, 2010.

[DLM+07] Ilias Diakonikolas, Homin K. Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco A. Servedio, and Andrew Wan. Testing for concise representations. *Proc. IEEE Symposium on Foundations of Computer Science*, 0:549–558, 2007.

[Fis01] Eldar Fischer. The art of uninformed decisions. *Bulletin of the EATCS*, 75:97, 2001.

[FKR+02] Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing juntas. In *FOCS*, pages 103–112, 2002.

[FM08] Eldar Fischer and Arie Matsliah. Testing graph isomorphism. *SIAM J. Comput.*, 38(1):207–225, 2008.

[FNS04] Eldar Fischer, Ilan Newman, and Jiří Sgall. Functions that have read-twice constant width branching programs are not necessarily testable. *Random Struct. Algorithms*, 24(2):175–193, 2004.

[FR87] P. Frankl and V. Rödl. Forbidden intersections. *Trans. Amer. Math. Soc. 300*, pages 259–286, 1987.

[FW81] P. Frankl and M. Wilson. Intersection theorems with geometric consequences. *Combinatorica 1*, pages 357–368, 1981.

[JPRZ04] Charanjit S. Jutla, Anindya C. Patthak, Atri Rudra, and David Zuckerman. Testing low-degree polynomials over prime fields. *Proc. IEEE Symposium on Foundations of Computer Science*, 0:423–432, 2004.

[KR04] Tali Kaufman and Dana Ron. Testing polynomials over general fields. In *Proc. IEEE Symposium on Foundations of Computer Science*, pages 413–422, Washington, DC, USA, 2004. IEEE Computer Society.

[KS05] Peter Keevash and Benny Sudakov. Set systems with restricted cross-intersections and the minimum rank of inclusion matrices. *SIAM J. Discrete Math.*, 18(4):713–727, 2005.

[PRS02] Michal Parnas, Dana Ron, and Alex Samorodnitsky. Testing basic boolean formulae. *SIAM J. Discrete Math.*, 16(1):20–46, 2002.

[SSS95] Jeanette P. Schmidt, Alan Siegel, and Aravind Srinivasan. Chernoff-Hoeffding bounds for applications with limited independence. *SIAM J. Discrete Math.*, 8(2):223–250, 1995.

A Definitions and tools from earlier works

Generalities. Let $n, k \in \mathbb{N}$ and $x \in \{0,1\}^n$. We use the following standard notation:

- $[n] = \{1, \ldots, n\}$ and $[k, n] = \{i \in [n] : k \leq i \leq n\}$;
- $|x| = |\{i \in [n] : x_i = 1\}|$;

Given a subset $I \subseteq [n]$, $x_{\restriction_I}$ denotes the restriction of x to the indices in I, and for $y \in \{0,1\}^{|I|}$, $x_{I \leftarrow y}$ denotes the string obtained by taking x and substituting its values in I with y (according to the natural ordering of $[n]$).

For a set S and $k \in \mathbb{N}$, $\binom{S}{k}$ is the collection of all k-sized subsets of S and $\binom{S}{\leq k}$ is the collection of all subsets of size at most k; a similar notation is used for binomial coefficients $\binom{m}{\leq k}$.

Given a pair $f, g : D \to \{0,1\}$ of Boolean functions, $\mathrm{dist}(f, g) \triangleq \Pr_{x \in D}[f(x) \neq g(x)]$. (Throughout this paper, $e \in S$ under the probability symbol means that an element e is chosen uniformly at random from a (multi)set S.) For a collection (property) $\mathcal{P}$ of functions $D \to \{0,1\}$, $\mathrm{dist}(f, \mathcal{P}) = \min_{g \in \mathcal{P}} \mathrm{dist}(f, g)$. For $\epsilon \in \mathbb{R}^+$, f is ϵ-far from $\mathcal{P}$ if $\mathrm{dist}(f, \mathcal{P}) \geq \epsilon$, otherwise it is ϵ-close to $\mathcal{P}$.

Influence, Juntas, Parities. For a function $g : \{0,1\}^n \to \{0,1\}$ and a set $A \subseteq [n]$, the *influence* of A on g is defined as

$$Inf_g(A) \triangleq \Pr_{x \in \{0,1\}^n,\ y \in \{0,1\}^{|A|}} \Big[g(x) \neq g(x_{A \leftarrow y}) \Big].$$

Note that when $|A| = 1$, this value is half that of the most common definition of influence of one variable; for consistency we stick to the previous definition instead in this case as well.

An index (variable) $i \in [n]$ is *relevant* with respect to g if $Inf_g(\{i\}) \neq 0$. A *k-junta* is a function g that has **at most** k relevant variables; equivalently, there is $S \in \binom{[n]}{k}$ such that $Inf_g([n] \setminus S) = 0$.

A *parity* is a linear form on $\mathbb{F}_2^n$. Such a linear $f : \{0,1\}^n \to \{0,1\}$ can be identified with a unique vector $v \in \{0,1\}^n$ such that $f(x) = \bigoplus_{i \in [n]} x_i v_i$ for all $x \in \{0,1\}^n$. We say that f is a *k-parity* if its associated vector has Hamming weight **exactly** k. The set of all k-parities will be denoted PAR_k.

Isomorphism testing. An *ϵ-tester* for f-isomorphism is a probabilistic algorithm $\mathcal{A}$ that, given oracle access to g, satisfies the following conditions: (1) if $f \cong g$ it accepts with probability at least 2/3; (2) if $\mathrm{distiso}(f, g) \geq \epsilon$ it rejects with probability at least 2/3. The query complexity of $\mathcal{A}$ is the worst-case number of queries it makes to g before making a decision. $\mathcal{A}$ is *non-adaptive* if its choice of queries does not depend on the outcomes of earlier queries. $\mathcal{A}$ has *one-sided error* if it always accepts in case $f \cong g$. By default, in all testers (and bounds) discussed in this paper we assume adaptivity and two-sided error, unless mentioned otherwise.

Note that testing f-isomorphism is equivalent to testing the property $\mathsf{Isom}_f \triangleq \{f^\pi : \pi \in G_n\}$ in the usual property testing terminology.

For any function f the query complexity for testing f-isomorphism is the query complexity of the best ϵ-tester for f-isomorphism. If $\mathcal{C}$ is a set of functions, then the query complexity for testing isomorphism to $\mathcal{C}$ is the maximum, taken over all $f \in \mathcal{C}$, of the query complexity for testing f-isomorphism.

Lemma for proving non-adaptive lower bounds. Let $\mathcal{P}$ be a property (subset) of functions mapping T to $\{0,1\}$. Define

$$\mathcal{R} \triangleq \{f \in \{0,1\}^T \mid \mathrm{dist}(f, \mathcal{P}) \geq \epsilon\}.$$

Any tester for $\mathcal{P}$ should, with high probability, accept inputs from $\mathcal{P}$ and reject inputs from $\mathcal{R}$.

We use the following lemma in various lower bound proofs for two-sided adaptive testing. It is proven implicitly in [FNS04], and a detailed proof appears in [Fis01]. Here we strengthen it somewhat, but still, the same proof works in our case too (we reproduce it here for completeness).

LEMMA A.1. *Let $\mathcal{P}, \mathcal{R}$ be as in the preceding discussion, and let D_Y and D_N be distributions over $\mathcal{P}$ and $\mathcal{R}$, respectively. If q is such that for all $Q \in \binom{T}{q}$ and $a \in \{0,1\}^Q$ we have*

$$\frac{2}{3} \Pr_{f \in D_Y}[f\restriction_Q = a] < \Pr_{f \in D_N}[f\restriction_Q = a],$$

then any tester for $\mathcal{P}$ must make more than q queries.

Proof. Assume towards a contradiction that there is such a tester making $\leq q$ queries; clearly we can assume it always makes exactly q queries. Define a distribution D obtained by selecting one of D_Y and D_N with probability $1/2$, and drawing an f from it. Fix a random seed so that the tester works for $f \in D$ with probability at least $2/3$; now the behaviour of the tester can be described by a deterministic decision tree of height q. Each leaf corresponds to a set $Q \in \binom{T}{q}$, along with an evaluation $a : Q \to \{0,1\}$; the leaf is reached if and only if f satisfies the evaluation. Consider the set S corresponding to accepting leaves; f is accepted if and only if there is $(Q,a) \in S$ such that $f\restriction_Q = a$. These $|S|$ events are disjoint, so the probability of acceptance of f is $\sum_{(Q,a)\in S} \Pr[f\restriction_Q = a]$.

Let $p = \Pr_{f\in D_Y}[f \text{ is accepted}]$, $r = \Pr_{f\in D_N}[f \text{ is rejected}]$. Now a standard averaging argument shows that $2/3p < r$, so $p - r < p/3 \leq 1/3$. The overall success probability when f is taken from D is $1/2 + (p-r)/2 < 2/3$, contradicting our assumption. $\square$