Electronic Colloquium on Computational Complexity, Report No. 93 (2010)

Nearly Tight Bounds for Testing Function Isomorphism

Sourav Chakraborty[*] David García-Soriano[*] Arie Matsliah[*]

Abstract

In this paper we study the problem of testing structural equivalence (isomorphism) between a pair of 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 can be queried.

We prove that for every $k \leq n$, the query complexity of testing isomorphism to k-juntas is $\Omega(k)$ and $O(k \log k)$. In particular, the (worst-case) query complexity of testing isomorphism to a given function $f : \{0,1\}^n \to \{0,1\}$ is $\widetilde{\Theta}(n)$.

Prior to our work, only lower bounds of $\Omega(\log k)$ queries were known, proved by Fischer et al. [FKR$^+$04], Blais and O'Donnell [BO10], and recently by Alon and Blais [AB10]. Our proof can also be extended to give polynomial query-complexity lower bounds for the problems of testing whether a function has a circuit of size $\leq s$, and testing whether the Fourier degree of a function is $\leq d$. This answers questions posed by Diakonikolas et al. [DLM$^+$07].

The nearly tight $O(k \log k)$ upper bound improves the $\widetilde{O}(k^4)$ upper bound from [FKR$^+$04] (and the similar bound that follows from [DLM$^+$07]). One implication of our techniques is a query-efficient procedure that given oracle access to any k-junta $g : \{0,1\} \to \{0,1\}$ can draw uniformly-random samples $(x, a) \in \{0,1\}^k \times \{0,1\}$ labelled by the core of g, each sample being correct with high probability. Generating such samples is one of the main ingredients of the testers from [DLM$^+$07]; while the procedure therein makes roughly k queries to g for obtaining each sample, our procedure requires only *one* query to g.

We also study the query complexity of testing isomorphism to k-juntas 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 obtained by proving that the query complexity of testing, with one-sided error, whether a function is a k-parity is $\Theta(\log \binom{n}{k})$.

Finally, we consider the problem of testing isomorphism between two unknown functions that can be queried. We prove that the (worst-case) query complexity in this setting is $\Omega(\sqrt{2^n})$ and $O(\sqrt{2^n n \log n})$.

[*] CWI, Amsterdam. Email: {sourav.chakraborty, david, ariem}@cwi.nl

ISSN 1433-8092

Contents

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*[1] *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).

On a wider perspective, answering Question 1.1, as also suggested by [FKR$^+$04] and [BO10], is an important step towards the meta-goal of characterizing testable properties of Boolean functions.

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 (for any $k \leq n$) can be tested with $O(1)$ queries follows from the work of Parnas et al. [PRS02].

Fischer et al. [FKR$^+$04] 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 (they actually prove more than that – a lower bound of $\Omega(\sqrt{k})$ for non-adaptive testers). Motivated by problems in machine learning, the focus on k-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$^+$04] with learning algorithms, Diakonikolas et al. [DLM$^+$07] developed a general 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$^+$04]. We elaborate more on [DLM$^+$07] and how it relates to our work in the following section.

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

Several related results, partially overlapping this work, were recently (and independently) obtained by Alon and Blais [AB10]. [AB10] proved that testing isomorphism to a known function

[1] Two functions are isomorphic if they are the same after some permutation of the variables.

requires $\Omega(n)$ non-adaptive queries. With $k = n$, our lower bound is asymptotically the same, and it works against adaptive testers as well. On the other hand, the lower bound in [AB10] is stronger in the sense that it applies to *most* functions $f : \{0,1\}^n \to \{0,1\}$. Alon and Blais also prove bounds similar to ours for the setting where both functions are unknown (see Appendix C).

2 Our results

2.1 Lower bounds

Our first result (Theorem 5.3) 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 certain k-juntas. Prior to our work, only lower bounds of $\Omega(\log k)$ queries were known [FKR+04, BO10].

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 $x, |x| \notin \frac{n}{2} \pm \sqrt{n}$; 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. As a corollary we obtain (see Corollary 5.4) an $\Omega(d)$ adaptive lower bound for testing if the Fourier degree of a Boolean function is at most d, improving on the non-adaptive bound of $\Omega(\log d)$ from [DLM+07]. (These bounds apply to testing the degree of Boolean f over *any* field. Better bounds are known for the case of finite fields; c.f. [AKK+03, JPRZ04, KR04]).

Furthermore, we prove that such an f can be quite restricted – it can be represented by a product of a threshold function and a polynomial of degree logarithmic in k; alternatively, it can be in $\mathcal{N}C/poly$. The latter property allows us to obtain lower bounds of $s^{\Omega(1)}$ queries for the problem of testing whether a function has a circuit of size s (see Theorem 6.1). This resolves one of the open problems from [DLM+07].

We remark that the fact that this indistinguishability result only holds for truncated random functions is essential – as Proposition D.1 in the Appendix says, random permutations of *any* k-junta f can be distinguished from completely random functions with $\widetilde{O}(\sqrt{k})$ queries and arbitrarily high constant success probability (also note that if the success probability is required to be only 3/4, a trivial such tester exists that makes only two queries).

2.2 Upper bounds

Our second result (Theorem 7.1) is a nearly matching upper bound of $O(k \log k)$ queries for testing isomorphism to any k-junta. Prior to this work, the only upper bounds known were roughly k^4 [FKR+04] (which also follows from [DLM+07]). One consequence of our techniques, which is of independent interest, is the following (see Proposition 7.15 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 $k \cdot \text{poly}(1/\epsilon)$ queries to g for obtaining each sample (while executing k independence tests of Fischer et al [FKR$^+$04]), our procedure requires only *one* query to g.

2.3 Testing isomorphism with one-sided error

Our third result (Theorem 8.1) 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$^+$04]. They actually proved that isomorphism against 2-juntas cannot be tested with one-sided error 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 $2 \leq k \leq n$, is between $\Omega(k \log(n/k))$ and $O(k \log n)$. (As we mentioned in the introduction, 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 $2 \leq k \leq n - 2$, is $\Theta(\log \binom{n}{k}) = \Theta(k \log(\frac{n}{\min\{k, n-k\}}))$. This seems an interesting result on its own, given the fact that 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 of 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 $O(\sqrt{n 2^n})$. As mentioned in the introduction, similar results for this setting were independently obtained by Alon and Blais [AB10].

2.5 Summary

In the following table we summarize our main results, and compare them to prior work.

	prior work	* n.a.	this work
testing isomorphism to k-juntas	$\Omega(\log k)$ [FKR$^+$04, BO10] $\widetilde{O}(k^4)$ [FKR$^+$04, DLM$^+$07]	$\Omega(\sqrt{k})$ [FKR$^+$04] $\Omega(k^{1/12})$ [BO10]	$\Omega(k)$ $O(k \log k)$
testing isomorphism to k-juntas with 1-sided error	$\Omega(\log \log n)$ [FKR$^+$04]	$\Omega(\log n)$ [FKR$^+$04]	$\Omega(k \log (n/k))$ $O(k \log n)$
testing the property of being a k-parity with 1-sided error			$\Theta(k \log(n/k))$
testing if a function can be computed by a circuit of size s	$\widetilde{\Omega}(\log s)$ [DLM$^+$07] $\widetilde{O}(s^6)$ [DLM$^+$07]		$s^{\Omega(1)}$
testing if a function has Fourier degree $\leq d$	$\Omega(\log d)$ [DLM$^+$07] $2^{O(d)}$ [DLM$^+$07]	$\Omega(\sqrt{d})$ [DLM$^+$07]	$\Omega(d)$
testing isomorphism of two unknown functions			$\Omega(\sqrt{2^n})$ $O(\sqrt{n2^n})$

In the second column (* n.a.) we mention the cases in which the general lower bounds were obtained via lower bounds for non-adaptive testers (in prior work). All our lower bounds apply to adaptive testers, and except for the $O(k \log k)$ upper bound for k-juntas, all upper bounds are obtained via non-adaptive one-sided error testers.

We also comment that nearly all our results extend to functions with general product domains and general ranges, following the lines of [DLM$^+$07] and [Bla09]. We defer this generalization to the full version of this paper.

3 Preliminaries and tools from earlier works

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

3.1 Generalities

If $f, g : T \to \{0,1\}$ are Boolean functions on some domain T, $\mathrm{dist}(f,g) \triangleq \Pr_{z \in T}[f(z) \neq g(z)]$. [2]

[2] 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 $W \subseteq [n]$, 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

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

For $V \subseteq [n]$, $x \in \{0,1\}^V$ and $w \in \{0,1\}^{[n]\setminus V}$ we define $x \sqcup w$ as the n-bit string y where $y_i = x(i)$ if $i \in V$ and $y_i = w(i)$ for $i \notin V$.

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 $x \in \{0,1\}^n$, $A \subseteq [n]$ and $y \in \{0,1\}^{|A|}$, we denote by $x_{A \leftarrow y}$ an input obtained by taking x and substituting its values in A with y (according to the natural ordering of $[n]$); we also define $x^{\oplus y} \triangleq x_{A \leftarrow (y \oplus (x\restriction_A))}$.

3.2 Permutations

The group $S_n = Sym([n])$ of all permutations of $[n]$ acts naturally on n-bit strings: permutation $\pi \in S_n$ sends $x = x_1 \ldots x_n \in \{0,1\}^n$ to $\pi(x) \triangleq x_{\pi(1)} \ldots x_{\pi(n)}$. Let $G_n \subseteq Sym(\{0,1\}^n)$ be the image of this action; $|G_n| = n!$. Given $f : \{0,1\}^n_W \to \{0,1\}$ and $\pi \in G_n$, we denote by f^π the function $f^\pi(x) \equiv f(\pi(x))$.

3.3 Property testing

For a collection (property) $\mathcal{P}$ of functions $T \to \{0,1\}$, $\text{dist}(f, \mathcal{P}) = \min_{g \in \mathcal{P}} \text{dist}(f, g)$. For $\epsilon \in \mathbb{R}^+$, f is ϵ-far from $\mathcal{P}$ if $\text{dist}(f, \mathcal{P}) \geq \epsilon$, otherwise it is ϵ-close to $\mathcal{P}$.

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 $g = f^\pi$. The *distance up to permutations of variables*, denoted by $\text{distiso}(f, g)$ is defined as $\min_{\pi \in G_n} \text{dist}(f^\pi, g)$. 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.

Notice that the problem of testing graph isomorphism in the dense graph model, as studied in [Fis05, FM08], is a special case of testing function isomorphism over the domain $\{0,1\}^n_2$. Similarly, if we restrict f and g to be functions from $\{0,1\}^n_k$ to $\{0,1\}$ then the problem is identical to testing k-uniform hypergraph isomorphism, as studied in [BC10].

3.4 Useful Lemma

Let $\mathcal{P}$ be a property (a subset) of the set $\{0,1\}^T$ of all functions from T to $\{0,1\}$. Define $\mathcal{R} \triangleq \{f \in \{0,1\}^T \mid \text{dist}(f, \mathcal{P}) \geq \epsilon\}$; any tester for property $\mathcal{P}$ 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 (see proof in Appendix B).

Lemma 3.1 *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 $a : Q \to \{0,1\}$ 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.

4 Brief overview of the proofs

4.1 Overview of the lower bounds

The proof of Theorem 5.3 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. For the proof of this part 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 G_n in our case is not transitive, which makes it harder to prove that a random permutation "shuffles" the values of a function uniformly. Another difference is that for the proof of Theorem 6.1 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 strict equality does not hold, e.g. for symmetric functions f, g satisfying $f(x) \equiv 1 - g(x)$.)

4.2 Overview of the upper bounds

The main ingredient in the proof of Theorem 7.1 is the nearly-optimal junta tester of Blais [Bla09]. Our 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$^+$04]. Then we perform the final test (the parallel of Occam's razor from [DLM$^+$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 (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 $2 \leq k \leq n/2$.

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, to obtain lower bounds of $\Omega(k \log(n/k))$ and $\Omega(k)$, respectively. The reason for this case distinction is to comply with the conditions of the combinatorial theorems.

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 $k' < k$) [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 $\leq q$ queries to accept g, even though it is $1/2$-far from any k-parity.

4.4 Overview of the remaining parts

The upper bound for testing with one-sided error (Proposition E.1), as well as the upper bound in the setting where both functions are unknown (Proposition C.1:Part 1 in the Appendix), is fairly straightforward. The testers start by random sampling, and then perform exhaustive search over all possible permutations, and check if one of them defines an isomorphism that is consistent with the samples. Their analysis is essentially the same as that of Occam's razor.

The lower bound in the setting where both functions are unknown (Proposition C.1:Part 2) is proved by defining two distributions on *pairs* of functions, the first supported on isomorphic pairs and the second on pairs that are far from being isomorphic. Then Yao's principle is applied via Lemma 3.1, which gives bounds on adaptive testers.

[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.

To prove that any function $f : \{0,1\}^n \to \{0,1\}$ is distinguishable from a completely random function (without the truncation) with $\widetilde{O}(\sqrt{n})$ queries (Proposition D.1) we borrow the ideas from [FM08], using which we reduce our problem to testing closeness of distributions, and then we apply the distribution tester of Batu et al. [BFF$^+$01].

5 Lower bound for testing isomorphism to k-juntas

Definition 5.1 *Let $\mathcal{F}_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$ denote the set of all functions $g : \{0,1\}^n \to \{0,1\}$ that satisfy $g(x) = 0$ for all $x, |x| \neq \frac{n}{2} \pm \lceil\sqrt{n}\rceil$; a* random truncated function *is a random function drawn from $\mathcal{F}_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil}$.*

Observation 5.2 *For any function $f : \{0,1\}^n \to \{0,1\}$ if $g : \{0,1\}^n \to \{0,1\}$ is a random truncated function then with high probability over the choice of g, $\text{distiso}(f,g) \geq \epsilon$ (for certain constant $\epsilon > 0$).*

Observe that to prove lower bounds for testing isomorphism to f, it suffices to show the stronger claim that there is a function g satisfying $\text{distiso}(f,g) \geq \epsilon$ and such that there is no algorithm that can distinguish, with high probability, between a random permutation of f and a random permutation of g. That is, given oracle access to a function h that is promised to be a random permutation of f or g (each with probability one half), no algorithm can determine which is the case. From Observation 5.2 we know that for any function f and a random truncated function g, with high probability, $\text{distiso}(f,g) \geq \epsilon$. So all we have to prove is the other part:

Theorem 5.3 *For any k there is a k-junta $f : \{0,1\}^n \to \{0,1\}$ such that for most truncated functions $g : \{0,1\}^n \to \{0,1\}$, any algorithm with oracle access to a function h will need at least $k - 5\log k$ queries in order to distinguish whether h is a random permutation of f or a random permutation of g. 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 circuits of size poly(k) and depth $O(\log k)$.*

Since 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 $b \triangleq n - \Theta(1)$, Theorem 5.3 implies the following:

Corollary 5.4 *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 $d \leq n - \omega(1)$.*

The proof of Theorem 5.3 is given in Section 5.3; it has two main parts.

In the first part we prove that there are "nice" Boolean functions f on k-bit strings, such that a random permutation of f is hard to distinguish from a random (truncated) function. This is

[4] The term "degree" here can refer 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 get the Fourier degree).

proved in Corollary 5.6 to Theorem 5.5. Theorem 5.5 itself is an intermediate result, that says the following: If we restrict our functions to be defined only on the $O(\sqrt{k})$ middle layers of the hypercube $\{0,1\}^k$, then there are "nice" functions that are hard to distinguish from a random function. Corollary 5.6 follows easily from Theorem 5.5: we extend these functions to the whole cube, by assigning zeroes outside the middle layers; then the corollary follows by observing that the number of inputs in the middle layers constitutes a constant fraction of all inputs, and that the zeroes outside the middle layers cannot help in the testing process. As we mentioned in the introduction, this step is essential (see Appendix D for a formal proof).

The second part of the proof of Theorem 5.3 uses a "preservation of distance under padding" argument (Lemma 5.14), which essentially allows us to embed a function on k variables into one on n variables, so that the hardness of testing remains roughly the same.

Theorem 5.5 *For any n there is a function $f : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$ such that for most functions $g : \{0,1\}^n_{\frac{n}{2}\pm\lceil\sqrt{n}\rceil} \to \{0,1\}$, any algorithm with oracle access to a function h will need at least $n - 5\log n$ queries in order to distinguish whether h is a random permutation of f or a random permutation of g. Moreover, f can have either of the following properties:*

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

To obtain Theorem 5.3 from Theorem 5.5 we use two different types of extensions. The first one extends functions defined on the truncated hypercube to functions on the whole hypercube. This gives us Corollary 5.6. The second extension takes us from Corollary 5.6 to Theorem 5.5. There we extend the dimension of a function by padding it, as described in Lemma 5.14.

Corollary 5.6 *For any k, there is a function $f : \{0,1\}^k \to \{0,1\}$ such that for most truncated functions $g : \{0,1\}^k \to \{0,1\}$, any algorithm with oracle access to a function h will need at least $k - 5\log k$ queries in order to distinguish whether h is a random permutation of f or a random permutation of g. 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 $\mathcal{NC}/poly$-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 $\{0,1\}^k$ that is zero outside $\{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}$. Let $K \triangleq |\{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil}| = \Theta(2^k)$ and let $\epsilon = \text{distiso}(f,g)K/2^k$ and assume there is an (q,ϵ)-tester of isomorphism to $ext(f)$. We can turn it into an $(q, \text{distiso}(f,g))$-tester of isomorphism to f in the obvious way: given access to $h : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$, test for isomorphism of $ext(h)$ to $ext(f)$. This works because $h = f$ implies $ext(h) = ext(f)$, and $\text{distiso}(h,f) \geq \epsilon 2^k/K$ implies $\text{distiso}(ext(h), ext(f)) \geq \epsilon$.

By Theorem 5.5, we have a $f : \{0,1\}^k_{\frac{k}{2}\pm\lceil\sqrt{k}\rceil} \to \{0,1\}$ that can either be represented by a degree $O(\log k)$ polynomial or by a $\mathcal{NC}/poly$-circuit. All we need to show is that $ext(f)$ has the required 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 $A, B \in \mathcal{NC}^1$, so the function $f'(x) = f(x) \wedge A(x) \wedge B(x)$ has the desired properties. ■

5.1 Central lemmas

As in [BC10], we show that there are functions $f : \{0,1\}^n_{\frac{n}{2} \pm \lceil\sqrt{n}\rceil} \to \{0,1\}$ that "look random" when restricted to small subsets of inputs, making Lemma 3.1 applicable.

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

Definition 5.7 *Let T be a finite domain and $r \in \mathbb{N}, \delta \in \mathbb{R}^+$. 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 5.8 *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 5.9 *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 $r \geq q$.

Consider any pair $\sigma_1, \sigma_2 \in G$ such that $\sigma_1(Q) \cap \sigma_2(Q) = \emptyset$. Since $2q \leq r$, 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 $r \geq n^3 q$. 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 5.10 *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 $x, y \in \{0,1\}^n_{\frac{n}{2} \pm \lceil \sqrt{n} \rceil}$,

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

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 $1 = \Pr_{\pi \in G}[\exists x \in Q, y \in V \mid \pi x = y] \leq \frac{q^2 s}{N}$, where we used the the union bound over x and y. Thus $s \geq \frac{N}{q^2}$. ■

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

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

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 5.11 (Chernoff bounds for k-wise independence [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 $0 \leq \delta \leq 1$,*

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

Since $ps \geq n^3$ and $k = n^3$, using Lemma 5.11 we obtain

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

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

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

$$\Pr_f[\exists\ Q, a \text{ such that } |A(f) - p| > p/5] \leq \binom{2^n}{q} 2^q n! 2^{-\Omega(n^3)} = o(1).$$

■

5.2 Proof of Theorem 5.5

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 simplicity 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 high probability, distiso$(f,g) = \Theta(1)$ is easily seen to hold, and also by Lemma 5.9 both functions are $(q, 1/5)$-regular.

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 $y \in D_Y$ has the desired property, whereas any $n \in D_n$ is distiso(f,g)-far from it. Let h be in the support of D_Y or D_N. Then h is also r-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 $a : Q \to \{0,1\}$,

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

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 3.1 establishes the main claim. Next we prove the two items in Theorem 5.5.

5.2.1 Proof of item 1 of Theorem 5.5

We need the following lemma, which gives us a n^4-uniform family of functions to draw f from, and this together with the argument above proves item 1.

Lemma 5.12 *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. [of Lemma 5.12] $\mathcal{F}_d$ 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 also been noted and 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. Thus any function $g : A\cup C \to \mathbb{F}$ can be evaluated by a polynomial 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. ■

5.2.2 Proof of item 2 of Theorem 5.5

We show that there are $(q,1/5)$-regular functions f that can be computed by small circuits. For this we need the following theorem:

Theorem 5.13 ([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 $O(\log(r\log B))$ 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 5.13 is n^4-independent. Furthermore, each f is computable with a bounded fan-in circuit of depth $O(\log(r\log B)) = O(\log n)$, and hence the circuit itself has size polynomial in n. Taking the closure of this family under G (considered as a multiset) we obtain a n^4-uniform family. By Lemma 5.9, there is a way to fix the *poly*(n) random bits so that the resulting function is $(n-O(\log n),1/5)$-regular.

5.3 Proof of Theorem 5.3

Let $k=k(n)\leq n$. Assume that there is a tester $\mathcal{A}$ that can test isomorphism against any $f:\{0,1\}^n\to\{0,1\}$ that is a k-junta with $o(k)$ queries. By Corollary 5.6, there is a function $f':\{0,1\}^k\to\{0,1\}$ that can be be computed by a $\mathcal{NC}/poly$-circuit or is representable by a product of a threshold function and a polynomial of degree $O(\log k)$, and such that given a random truncated function $g':\{0,1\}^k\to\{0,1\}$, any adaptive ϵ-tester with oracle access to g' will require $k-O(\log k)$ queries to distinguish a random permutation of f' from a random permutation of g'.

Let $f:\{0,1\}^n\to\{0,1\}$ be the extension of f', where $f(x)=f'(x\restriction_{[k]})$ for all $x\in\{0,1\}^n$. From Lemma 5.14 if follows that if g is the extension of a random truncated function $g':\{0,1\}^k\to\{0,1\}$ then it is impossible for any algorithm to distinguish, using fewer than q queries, a random permutation of f from a random permutation of g.

Lemma 5.14 *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 $f=ext(f')$ to be the 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 $g=ext(g')$. Then the following holds:*

- *If* $\mathrm{distiso}(f', g') \geq \epsilon$, *for some* $\epsilon \in \mathbb{R}^+$ *then* $\mathrm{distiso}(f, g) \geq \epsilon/4$.
- *If is impossible for any algorithm, using only q queries, to distinguish a random permutation of f' from a random permutation of g' then it is impossible for any algorithm to distinguish using fewer than q-queries a random permutation of f from a random permutation of g.*

Proof. Assume that $\mathrm{distiso}(f, g) < \epsilon/4$, that is, $\mathrm{dist}(f, g^\pi) < \epsilon/4$ for some $\pi \in G$. We prove that this implies $\mathrm{distiso}(f', g') < \epsilon$.

Let $A = \pi([k]) \setminus [k]$. Let $I(A) \triangleq Inf_{g^\pi}(A)$ denote the influence of g^π in A. First, observe that $I(A) \leq 2 \cdot dist(f, g^\pi) < \epsilon/2$, due to the fact that f does not depend on the indices in A at all. This is easy to check; see also Lemma 7.9.

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

$$\pi'(i) = \left\{ \begin{array}{lcl} \pi(i) & , & i \in [k] \text{ and } \pi(i) \in [k] \\ \sigma(\pi(i)) & , & i \in [k] \text{ and } \pi(i) \in A \\ \sigma(i) & , & i \in [k] \setminus A \end{array} \right\}.$$

Informally, π' is obtained from π by "bringing back" to $[k]$ all those $i \in [k]$ that were mapped to A. From $I(A) < \epsilon/2$ we get $\mathrm{dist}(g^\pi, g^{\pi'}) < \epsilon/2$, and by the triangle inequality, $\mathrm{dist}(f, g^{\pi'}) < \epsilon$. But since π' maps all relevant indices in $[k]$ to $[k]$, it defines a valid permutation of $[k]$, therefore, $\mathrm{dist}(f', g'^{\pi'}) = \mathrm{dist}(f, g^{\pi'}) < \epsilon$.

It is clear that if $f' \cong g'$ then $f \cong g$. Let there be an algorithm $\mathcal{A}$ that can distinguish 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 $ext(h)^\sigma$ (clearly, any query to h can be simulated by one query to $ext(h)^\sigma$, and the distribution of $ext(h)^\sigma$ is a random permutation of either f or g). Hence no such $\mathcal{A}$ exists. ■

6 Lower bounds for testing size-s Boolean circuits

Theorem 6.1 *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 5.6, 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 5.6) 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 $r = \Theta(s^{1/c})$. Now define $f : \{0,1\}^n \to \{0,1\}$ to be the extension of f', where $f(x) = f'(x\restriction_{[k]})$ for all $x \in \{0,1\}^n$. From Lemma 5.14 we obtain that given $g : \{0,1\}^n \to \{0,1\}$ which is the extension of a random truncated function g' any algorithm that makes $o(r)$ 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 $\Omega(r) = \Omega(s^{1/c})$. ■

7 Upper bound of $O(k \log k)$ for testing isomorphism to k-juntas

Theorem 7.1 *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 1 below). Informally, RobustIsoTest allows us to test isomorphism between a known function f and an unknown g, 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 $\le k$ 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.

7.1 Testing isomorphism tolerantly with noise

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

Definition 7.2 *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 $A \subseteq \{0,1\}^k$, $\left|\Pr_{x\sim\mathcal{D}}[x \in A] - |A|/2^k\right| \le \mu$;*
- *$\Pr[a = g(x)] \ge 1 - \eta$,*

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 7.3 *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 $\widetilde{g}$ for some $g : \{0,1\}^k \to \{0,1\}$, where $\eta \le \epsilon/100$ and $\mu \le \epsilon/10$, 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 1. It is clear that RobustIsoTest uses $O(\frac{k\log k}{\epsilon^2})$

Algorithm 1 (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 $\pi$ of $[k]$ such that $\left|\left\{i \in [q] : f^\pi(x^i) \ne a^i\right\}\right| < \epsilon q/2$.
```

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

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

Using Chernoff bounds (additive form) we have

$$\Pr\left[\left||\{i \in [q] : f^\pi(x^i) \ne g(x^i)\}| - \zeta_\pi q\right| > \epsilon q/10\right] = 2^{-\Omega(\epsilon^2 q)},$$

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

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

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:

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

Therefore, with probability at least 9/10,

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

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, $\delta_{\pi} q - 2\epsilon q/5 > \epsilon q/2$. ■

7.2 Useful definitions and lemmas

Definition 7.4 *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 7.5 (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 7.6 (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 7.7 (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. *For every $\mathcal{I}$ and $\mathcal{J} \subseteq \mathcal{I}$, the infinity distance 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 7.6 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 $k = |\mathcal{J}|$. Assume $\ell > 4k^2$; otherwise the claim is trivial. We may also assume 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

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

and

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

Using the inequalities $1 - my \leq (1-y)^m$ for all $y < 1, m \in \mathbb{N}$ and $(1+y)^m \leq e^{my} \leq 1 + 2my$ for all $m \geq 0, 0 \leq my \leq 1/2$, 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 7.8 (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.

7.3 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 7.9 *[FKR$^+$04] For any $f : \{0,1\}^n \to \{0,1\}$ and $A \subseteq [n]$*

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

We will also use the fact (see [FKR$^+$04, Bla09] for a proof) that influence is monotone and subadditive; namely, for all $f : \{0,1\}^n \to \{0,1\}$ and $A, B \subseteq [n]$,

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

For the following definition and lemma we recall the distributions $\mathcal{D}_{\mathcal{I}}$ and $\mathcal{D}_{\mathcal{J}}$ from Definition 7.6.

Definition 7.10 *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 $\mathcal{I}$, 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 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 7.11 *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 7.10, 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 7.10 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) $g^*(y) = \mathsf{core}_k(g^*)(x')$,

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

The first equality is by Definition 7.5, and the second one follows from items 1a and 3a in Definition 7.10. From item 1b we also have

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

where r is picked uniformly at random. However, by the second item of Lemma 7.7, the distribution $\mathcal{D}_{\mathcal{J}}$ is $4k^2/\ell$ close to uniform [6]; combining this with (3) we also get

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

Likewise, we have

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

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

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

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. ■

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

Corollary 7.12 *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 7.7. The second item follows from Lemma 7.11. ■

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 7.13 (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 7.10). (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 $\mathrm{T}_{[\mathrm{Bla09}]}$ works. Given the partition $\mathcal{I}$, $\mathrm{T}_{[\mathrm{Bla09}]}$ starts with an empty set $S = \emptyset$, and iteratively finds indices $j \in [\ell] \setminus S$ such that for some pair of inputs $y, y' \in \{0,1\}^n$, $y_{\restriction_{[n]\setminus I_j}} = y'_{\restriction_{[n]\setminus I_j}}$ 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. $\mathrm{T}_{[\mathrm{Bla09}]}$ stops at some stage, and rejects if and only if $|S| > k$. The Main Lemma in [Bla09] asserts that if g is not rejected (i.e. if $\mathrm{T}_{[\mathrm{Bla09}]}$ terminates with $|S| \leq k$), then

[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.

$(*)$ 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 7.13. 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 $\mathrm{T}_{[\mathrm{Bla09}]}$ did not reject. In this case,

- by $(*)$, with probability at least 19/20 the set $\mathcal{J}$ satisfies
$$Inf_g\Big([n] \setminus (\bigcup_{I_j\in\mathcal{J}} I_j)\Big) \leq Inf_g\Big([n] \setminus (\bigcup_{j\in S} I_j)\Big) \leq \epsilon/4800;$$
- 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 7.7, $\Pr_{\mathcal{I},y\sim\mathcal{D}_\mathcal{I}}\Big[g(y) = g^*(y)\Big] = \mathrm{dist}(g, g^*)$; hence by Markov's inequality, with probability at least 9/10 the partition $\mathcal{I}$ satisfies $\Pr_{y\sim\mathcal{D}_\mathcal{I}}[g(y) \neq g^*(y)] \leq 10 \cdot \mathrm{dist}(g, g^*)$.

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 $\mathrm{T}_{[\mathrm{Bla09}]}$. Since $\mathrm{dist}(g, g^*) \leq \epsilon/2400$, we have $Inf_g([n] \setminus V) \leq \epsilon/1200$ (by Lemma 7.9). By the subadditivity and monotonicity of influence we get

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

where the second inequality follows from $V \setminus U \subseteq [n] \setminus (\bigcup_{I_j\in\mathcal{J}} I_j)$. This means, by Lemma 7.9, 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 7.10. ■

7.4 The final algorithm

Consider the tester described in Algorithm 2. The proof of Theorem 7.1 follows from the next lemma:

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

Algorithm 2 (tests isomorphism to a k-junta f)

```
1: let $\ell = \ell_{[\text{Bla09}]}(k, \epsilon) = \Theta(k^9/\epsilon^5)$
2: randomly partition $[n]$ into $\mathcal{I} = (I_1, \dots, I_\ell)$
3: test $g$ for being a $k$-junta, using $\mathrm{T}_{[\text{Bla09}]}$ with $\mathcal{I} = I_1, \dots, I_\ell$ (see Theorem 7.13)
4: if $\mathrm{T}_{[\text{Bla09}]}$ rejects then
5:     reject
6: end if
7: let $\mathcal{J} \subseteq \mathcal{I}$ be the set output by $\mathrm{T}_{[\text{Bla09}]}$
8: construct $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ (see Section 7.2)
9: accept if and only if RobustIsoTest($\mathsf{core}_k(f), \mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$) accepts (see Section 7.1)
```

Lemma 7.14 *Algorithm 2 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 2 does not reject on line 5, because $\mathrm{T}_{[\text{Bla09}]}$ has one-sided error. So in this case, by Theorem 7.13, with probability at least $4/5$ the pair $(\mathcal{I}, \mathcal{J})$ is $\epsilon/600$-good. If so, by Corollary 7.12, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ 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 $\mathrm{T}_{[\text{Bla09}]}$ rejects with probability greater than $2/3$ and we are done. So assume that the second case holds.

By the third item of Theorem 7.13, with probability at least $4/5$, $\mathrm{T}_{[\text{Bla09}]}$ either rejects g, or the pair $(\mathcal{I}, \mathcal{J})$ is $\epsilon/600$ good. If $\mathrm{T}_{[\text{Bla09}]}$ rejects then we are done. Otherwise, if an $\epsilon/600$-good pair is obtained, then by Corollary 7.12, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ 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 $\mathrm{T}_{[\text{Bla09}]}$, and additional $O(k \log k/\epsilon^2)$ queries made by RobustIsoTest.

This completes the proof of Theorem 7.1.

7.5 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 7.15 *Let $\gamma > 0$ be an arbitrary constant. There is a randomized algorithm A, that given oracle access to any k-junta $g : \{0,1\}^n \to \{0,1\}$ does the following:*

Operation: *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 $(x,a) \in \{0,1\}^k \times \{0,1\}$.*

Performance: *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 $\mathrm{T}_{\text{[Bla09]}}$ 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 $\mathrm{T}_{\text{[Bla09]}}$ (see Theorem 7.13).

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 7.13 (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 7.12, $\mathsf{sampler}_{\mathcal{I},\mathcal{J}}(g)$ 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 $\Pr[a \neq g(x)] \leq \epsilon/100 + \mu/(1+\mu) < \epsilon/50 < \gamma$. ■

8 Testing isomorphism with one-sided error

Note that if $f \in \mathsf{PAR}_k$, then testing isomorphic to f is the same as testing membership in PAR_k. The main theorem in this section is the following:

Theorem 8.1 *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 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 $\Theta(1)$.*

We start 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 8.2 *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.*

It is also easy to verify that the second item of Theorem 8.1 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 we only have to prove the first item of Theorem 8.1 for $k \in [2, \lceil n/2 \rceil]$.

In Section 8.1 we prove the lower bound for the first item of Theorem 8.1. The upper bound for the first item of Theorem 8.1 follows immediately from Proposition E.1 proved in Appendix E.

8.1 Lower bound for testing isomorphism to k-parities with one-sided error

To prove the lower bound, 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 Section 4.3.

8.1.1 Lower bound of $\Omega(\log n)$ for $2 \leq k \leq \lceil n/2 \rceil$

Let $q = \log\lceil n/2 \rceil + 2$, and let $x^1, \ldots, x^q \in \{0,1\}^n$ be the set of queries. For any $k \in [2, \lceil n/2 \rceil]$ 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 $i \in [q]$; 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.

8.1.2 Lower bound of $\Omega(\log\binom{n}{k})$ for $5 \leq k \leq \alpha n$

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

Theorem 8.3 ([FW81]; see also Frankl-Wilson Theorem in [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 $F, F' \in \mathcal{F}$, $|F \cap F'| \neq \ell/2$, then $|\mathcal{F}| \leq \binom{m}{\ell/2}\binom{3\ell/2-1}{\ell}/\binom{3\ell/2-1}{\ell/2}$.*

Let $q = \lfloor \frac{1}{20}\log\binom{n}{k} \rfloor$. Given $k \in [5, \lfloor n/2 \rfloor]$, let $k' \geq 1$ be the smallest integer such that $(k-k')/2$ is a power of a prime; note that $k' < k/2$ as $k \geq 5$. 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 $x^1, \ldots, x^q \in \{0,1\}^n$ there exists $y \in \{0,1\}^n$ such that

- $|y| = k - k'$,
- $y \cap g = \emptyset$ and
- $< y, x^i > \triangleq \bigoplus_{j=1}^{n}(y_j \cdot x_j^i) = 0$ 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 $x^1, \ldots, x^q$.

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 $< y, x^i >= \alpha_i$ 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 $< y, x^i >= 0$ for all $i \in [q]$.

Let $c \triangleq n/k$; it is easy to verify that $c \leq m/\ell \leq 2c$. In the following we use the bounds $b(\log a - \log b) \leq \log\binom{a}{b} \leq b(\log a - \log b + 2)$. We have

$$\log|Y_\alpha| \geq \log\Big(\frac{\binom{n-k'}{k-k'}}{2^q}\Big) = \log\binom{m}{\ell} - \frac{1}{20}\log\binom{n}{k} \geq$$

$$\geq \ell(\log m - \log \ell) - \frac{1}{20}k(\log n - \log k + 2) \geq \ell(\log c - 1) - \frac{1}{10}\ell(\log c + 2) = \ell(\frac{9}{10}\log c - \frac{6}{5}).$$

On the other hand,

$$\log\Big(\frac{\binom{m}{\ell/2}\binom{3\ell/2-1}{\ell}}{\binom{3\ell/2-1}{\ell/2}}\Big) \leq \frac{\ell}{2}(\log m - \log \ell + 3) + 3\ell/2 \leq \ell(\frac{1}{2}\log c + \frac{7}{2}).$$

Since $c \geq 2^{12}$, these inequalities together with Theorem 8.3 imply that there must be $y^1, y^2 \in Y_\alpha$ such that $|y^1 \cap y^2| = \ell/2$, as required.

8.1.3 Lower bound of $\Omega(k)$ for $\alpha n \leq k \leq \lceil n/2 \rceil$

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 8.4 ([FR87]) *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 $|\mathcal{F}| \leq 2^{(1-\delta)2k}$.*

Let n be large enough with respect to α and δ. Given $k \in [\alpha n, \lceil n/2 \rceil]$, 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 $x^1, \ldots, x^q \in \{0,1\}^n$ there exists $y \in \{0,1\}^n$ such that

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

Let $Y = \{y \in \{0,1\}^n : y \subseteq [2k] \cap [n] \text{ and } |y| = k\}$. 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 $< y, x^i >= \alpha_i$ for all $i \in [q]$. One of the sets Y_α must be of size at least $\binom{2k-1}{k}/2^q$, 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 8.4, 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 $< y, x^i >= 0$ 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 $< y, x^i >= 0$ 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.

Acknowledgement We thank Ronald de Wolf for many useful discussions.

References

[AB10] Noga Alon and Eric Blais. Testing boolean function isomorphism. *Personal communication*, 2010.

[AKK+03] Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, Dana Ron, and Dana. Testing low-degree polynomials over gf(2). In *Proceedings of 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 *APPROX-RANDOM*, pages 366–377, 2009.

[BFF$^+$01] Tugkan Batu, Lance Fortnow, Eldar Fischer, Ravi Kumar, Ronitt Rubinfeld, and Patrick White. Testing random variables for independence and identity. In *FOCS*, pages 442–451, 2001.

[Bla09] Eric Blais. Testing juntas nearly optimally. In *STOC*, pages 151–158, 2009.

[BO10] Eric Blais and Ryan O'Donnell. Lower bounds for testing function isomorphism. In *IEEE Conference on Computational Complexity (to appear)*, 2010.

[DLM$^+$07] Ilias Diakonikolas, Homin K. Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco A. Servedio, and Andrew Wan. Testing for concise representations. In *FOCS*, pages 549–558, 2007.

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

[Fis05] Eldar Fischer. The difficulty of testing for isomorphism against a graph that is given in advance. *SIAM J. Comput.*, 34(5):1147–1158, 2005.

[FKR$^+$04] Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing juntas. *J. Comput. Syst. Sci.*, 68(4):753–787, 2004.

[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. *Foundations of Computer Science, Annual IEEE Symposium on*, 0:423–432, 2004.

[KR04] Tali Kaufman and Dana Ron. Testing polynomials over general fields. In *FOCS '04: Proceedings of the 45th Annual 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 Notation

A.1 Generalities

Let $n, k \in \mathbb{N}$, $x, y \in \{0,1\}^n$ and $f, g : \{0,1\}^n \to \{0,1\}$. 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. If D, R are two sets then R^D denotes the set of all functions from D to R.

A.2 Boolean functions

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

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

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.

A.3 Property 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 $\text{distiso}(f, g) \geq \epsilon$ 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.

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.

B Proof of Lemma 3.1

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 leave corresponds to a set $Q \in \binom{T}{q}$, along with an evaluation $a : Q \to \{0,1\}$; the leave 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 $p = \Pr_{f \in D_Y}[f \text{ is accepted}]$, $q = \Pr_{f\in D_N}[f \text{ is accepted}]$. Now a standard averaging argument shows that $2/3p < q$, so $p - q < p/3 \leq 1/3$. The overall success probability when f is taken from D is $1/2 + (p-q)/2 < 2/3$, contradicting our assumption.

C Testing in the unknown-unknown setting

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 $\text{distiso}(f,g) \geq \epsilon$ 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 C.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 $\Omega(2^{n/2})$.*

C.1 Proof of Proposition C.1 Part 1: Upper Bound

In this section we show that isomorphism between 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 3.

It is clear that Algorithm 3 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

Algorithm 3 (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 $\sqrt{\frac{2n\log n}{\epsilon 2^n}}$, 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 $\pi$ such that for all $x \in Q$ either $f(x) = g(\pi(x))$ or $\pi(x) \notin Q$
```

$\pi(x)$ are in Q is at least $\frac{2n\log n}{\epsilon 2^n}$, so any such π passes the acceptance condition with probability at most $(1 - 2n\log n/(\epsilon 2^n))^{\epsilon 2^n} \leq e^{-2n\log n}$. The proof follows by taking the union bound over all $n!$ permutations.

C.2 Proof of Proposition C.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 function drawn according to distribution D_Y are isomorphic, while any pair drawn according to distribution D_N is 1/8-far from isomorphic.

We define an almost-random function $f : \{0,1\}^n \to \{0,1\}$ 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 an almost-random 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 almost-random functions f and g. Now with probability $1 - o(1)$ the two functions are 1/8 far from each other. So we can assume that that the pairs obtained by D_N are of functions that are 1/8 far from each other.

For any $Q = \{x_1, \dots, x_t\} \subset \{0,1\}^n$ 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 $\Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)]$. We show that for any $Q \subset \{0,1\}^n$, if $|Q| < 2^{n/2}/10$, then for any $p, q \in \{0,1\}^t$ we have

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

This implies (by Lemma 3.1) a lower bound of $2^{n/2}/10$ on the adaptive query complexity for the two-sided testing for the unknown-unknown setting.

Since any function f generated by D_Y and D_N has the property that $f(x) = 0$ if $|x|$ is less than $\frac{n}{2} - \sqrt{n}$ or greater than $\frac{n}{2} + \sqrt{n}$, we can assume that for all $x^i \in Q$ the number of 1's in each is between $n/2 - \sqrt{n}$ and $n/2 + \sqrt{n}$.

Let Q be $\{x^1, \ldots, x^t\}$. Clearly, if the pair is drawn according to distribution D_N then the answers to the queries will be uniformly distributed. So 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. For all i, let $\pi(x^i)$ denote the string obtained by permuting the bits of x^i according to the permutation π. Let E_Q denote the event that for all i,j we have $\pi(x^i) \neq x^j$. Conditioned on the event E_Q the answers to the queries will again be distributed uniformly, that is

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

Claim C.2 *With probability at least $2/3$ event E_Q occurs, that is, for all i,j we have $\pi(x^i) \neq x^j$.*

Proof. [Of Claim C.2] 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. ∎

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

$$(\Pr[E_Q]) \Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)] + (1 - \Pr[E_Q]) \Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q)].$$

Using the above claim we have $\Pr_{(f,g)\in D_Y}[(f,g)\restriction_Q = (p,q)] \geq (2/3)\Pr_{(f,g)\in D_N}[(f,g)\restriction_Q = (p,q)]$. This implies a lower bound of $2^{n/2}/10$ on the adaptive query complexity for the two-sided testing for the unknown-unknown setting.

D 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 D.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.*

We prove the above proposition momentarily, but first we 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 the singletons (similarly, 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.

Proof. 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 graphs 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. We work out the formal details below.

Let $\ell = 2\log n$. Given a function $f : \{0,1\}^n \to \{0,1\}$ and $i \in [n]$ we define $\alpha(f, i) \in \{0,1\}^\ell$ as follows: the j'th bit of $\alpha(f, i)$ is one if and only if $f(\{i, j\}) = 1$. We then define the distribution D_f over $\{0,1\}^\ell$, where the probability of $\beta \in \{0,1\}^\ell$ under D_f is $\frac{1}{n}|\{i \in [n] : \alpha(f, i) = \beta\}|$. Clearly, if $f = g$ then $D_f = D_g$. Now we claim something similar for f and g that are isomorphic.

Let Π be a set of permutations of $[n]$, such that there is one-to-one correspondence between the elements of Π and the possible injections $I : [\ell] \to [n]$ as follows. Each $\pi \in \Pi$ is associated with an injection $I_\pi : [\ell] \to [n]$, such that

$$\pi(i) = \left\{ \begin{array}{lcl} I_\pi(i) & , & i \in [\ell] \\ i & , & i \in [n] \setminus [\ell] \text{ and } I_\pi^{-1}(i) = \emptyset \\ I_\pi^{-1}(i) & , & i \in [n] \setminus [\ell] \text{ and } I_\pi^{-1}(i) \neq \emptyset \end{array} \right\}.$$

Clearly, $|\Pi| \leq n^\ell$.

Claim D.2 *If f is isomorphic to g, then for some $\pi \in \Pi$, $D_f = D_{g^\pi}$. On the other hand, for any function f,*

$$\Pr_g \left[|D_f - D_{g^\pi}| \leq 1/4 \textit{ for some } \pi \in \Pi \right] = 1 - o(1).$$

[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.

Proof. The first statement is straightforward: Let f and g be isomorphic, i.e. $f = g^{\sigma}$ for some $\sigma : [n] \to [n]$. Take $\pi \in \Pi$ such that $\sigma(i) = \pi(i)$ for all $i \in [\ell]$. Then $D_f = D_{g^{\pi}}$.

Now, fix f, and let g be chosen uniformly at random. We would like to show that for all $\pi \in \Pi$, $\Pr_g \Big[|D_f - D_{g^{\pi}}| \leq 1/4\Big] = 1 - o(1/|\Pi|)$, so that we can apply the union bound. But notice that it is sufficient to prove this inequality when π is the identity, because the function g is chosen uniformly at random.

Fix $i \in [n]$. For every $j \in [n]$,

$$\Pr_g \Big[\alpha(f, i) = \alpha(g, j)\Big] = 2^{-\ell},$$

hence

$$\Pr_g \Big[\alpha(f, i) = \alpha(g, j) \text{ for some } j \in [n]\Big] \leq n2^{-\ell} = 1/n.$$

Therefore, the expected intersection size between the multisets[10] $\{\alpha(f, i) : i \in [n]\}$ and $\{\alpha(g, i) : i \in [n]\}$ is $O(1)$. But notice that in order for the distributions D_f and D_g to be close, the intersection of these multisets must be of size $\Omega(n)$. Using the fact that the events

$$E_i \triangleq \mathbb{I}\Big[\alpha(f, i) = \alpha(g, j) \text{ for some } j \in [n]\Big]$$

are independent, we can apply standard concentration bounds to conclude that

$$\Pr_g \Big[|D_f - D_g| \leq 1/4\Big] = 1 - 2^{-\Omega(n)} = 1 - o(1/|\Pi|),$$

completing the proof. ■

Notice that the distribution D_f can be constructed exactly given f. On the other hand, given an oracle access to g, we can obtain a random sample from D_g by picking a random $i \in [n]$ and querying g on ℓ inputs $\{i, 1\}, \ldots, \{i, \ell\}$. This observation, together with Claim D.2, suggests that we use the following lemma from [BFF$^+$01], which states that $\widetilde{O}(\sqrt{n})$ samples are sufficient for testing equivalence between a samplable distribution and an explicitly given one.

Lemma D.3 *There is a tester T_{Dist} that for any two distributions D_K,D_U over $\{0, 1\}^*$, each having support of size at most n, and where D_K is given explicitly and D_U is given as a black box that allows sampling, satisfies the following: If $D_K = D_U$ then the T_{Dist} accepts with probability at least $1 - n^{-3\log n}$; and if $|D_K - D_U| \geq 1/4$ then T_{Dist} rejects with probability at least $1 - n^{-3\log n}$. In any case, T_{Dist} uses $\widetilde{O}(\sqrt{n})$ samples.*

Actually, this is an amplified version of the lemma from [BFF$^+$01], which can be achieved by independently repeating the algorithm provided there polylog(m) many times and taking the majority vote.

To conclude, we can reduce our problem to testing equivalence of distributions as follows. Given f and oracle access to g, go over all permutations $\pi \in \Pi$ and test, with T_{Dist}, if D_f and $D_{g^{\pi}}$ are equal. If T_{Dist} accepts for some π, accept; otherwise reject.

[10] Intersection here can be a multiset as well. For example, $\{a, a, b, c, c, c\} \cap \{a, a, b, b, c\} = \{a, a, b, c\}$.

By Claim D.2, if f is isomorphic to g then for some $\pi \in \Pi$ we have $D_f = D_{g^\pi}$, and so T_{Dist} will with high probability accept while checking that particular π. On the other hand, every π for which $|D_f - D_{g^\pi}| \geq 1/4$ is accepted with probability at most $n^{-3 \log n} = o(1/|\Pi|)$. Thus, for randomly chosen g, T_{Dist} rejects with probability $1 - o(1)$.

As for the query complexity, the amplified version of Lemma D.3 allows us to reuse the same $\widetilde{O}(\sqrt{n})$ samples for checking all permutations in Π. Therefore, since simulating a random sample from D_{g^π} requires $\ell = 2 \log n$ queries to g, the bound on the query complexity is $\widetilde{O}(\sqrt{n})$.

■

E Upper bound for testing isomorphism to k-juntas with one-sided error

Recall that for every $f : \{0,1\}^n \to \mathcal{R}$, Isom_f denotes the collection of functions isomorphic to f.

Proposition E.1 *Isomorphism to any given $f : \{0,1\} \to \mathcal{R}$ can be tested with $O(\log |\mathsf{Isom}_f|/\epsilon)$ queries.*

This immediately implies the desired upper bound, since $|\mathsf{Isom}_f| \leq \binom{n}{k} \cdot k!$ for any $k \in [n]$ and k-junta f. In view of Theorem 8.1, this implies a tight upper bound for testing isomorphism against k-parities, since $|\mathsf{PAR}_k| = \binom{n}{k}$.

Proof. [of Proposition E.1] Consider the simple tester described in Algorithm 4. It is clear that

Algorithm 4 (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 $i \in [q]$
```

this is a non-adaptive one-sided error tester, and that for any fixed $\epsilon > 0$ it makes only $O(\log |\mathsf{Isom}_f|)$ 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 $|\mathsf{Isom}_f|(1-\epsilon)^q \leq |\mathsf{Isom}_f| e^{-\epsilon q} < 1/3$. ■

In addition to the fact that Proposition E.1 implies a tight upper bound of $O(\log \binom{n}{k})$ queries for testing PAR_k, something much stronger holds. Since the distance between any two parity functions is 1/2, the algorithm from Proposition E.1 (which can be thought of as a learning algorithm) can actually *decode* the parity bits of the tested function with the same number of queries:

Fact E.2 *There is a non-adaptive algorithm A that, given n,k and oracle access to $g : \{0,1\}^n \to \{0,1\}$, satisfies the following:*

- *if g is a k-parity then A outputs the k parity indices of g with probability* 1*;*
- *if g is ϵ-far from being k-parity then A rejects with probability at least* 2/3*;*
- *A makes $O(\log \binom{n}{k})$ queries to g.*

Furthermore, without the second item, A can be even made deterministic[11].

This is in contrast to the matching lower bound that applies even for the much simpler task of deciding whether the size of a given parity is k.

[11] The fact that for all n and k there is such deterministic algorithm follows from a simple probabilistic argument.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de