2011 26th Annual IEEE Conference on Computational Complexity

Property Testing Lower Bounds Via Communication Complexity

Eric Blais
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA
eblais@cs.cmu.edu

Joshua Brody*
IIIS, ITCS
Tsinghua University
Beijing, China
joshua.e.brody@gmail.com

Kevin Matulef*
IIIS, ITCS
Tsinghua University
Beijing, China
matulef@gmail.com

***Abstract*—We develop a new technique for proving lower bounds in property testing, by showing a strong connection between testing and communication complexity. We give a simple scheme for reducing communication problems to testing problems, thus allowing us to use known lower bounds in communication complexity to prove lower bounds in testing. This scheme is general and implies a number of new testing bounds, as well as simpler proofs of several known bounds.**

For the problem of testing whether a boolean function is k-linear (a parity function on k variables), we achieve a lower bound of $\Omega(k)$ queries, even for adaptive algorithms with two-sided error, thus confirming a conjecture of Goldreich [25]. The same argument behind this lower bound also implies a new proof of known lower bounds for testing related classes such as k-juntas. For some classes, such as the class of monotone functions and the class of s-sparse $GF(2)$ polynomials, we significantly strengthen the best known bounds.

I. Introduction

The field of property testing seeks to formalize the question: what can we determine about a large object, with limited access to the object itself? In general the large object may by anything—for instance a graph on n nodes, or a function on n variables. In a typical property testing setup, a tester who has unbounded computational power is given query access to the large object. The tester's goal is to accept the object if it has some property $\mathcal{P}$, and reject it if it is "far" from having property $\mathcal{P}$.

In this paper we will primarily concern ourselves with the case when the large object is a boolean function f on n bits. In this case, the tester's goal is to accept f with probability at least $2/3$ if f has property $\mathcal{P}$, and reject with probability at least $2/3$ if f must be modified on an ϵ fraction of the 2^n possible inputs in order to have property $\mathcal{P}$. The query complexity (i.e. the number of times the testing algorithm must query f) should hopefully be a small function of ϵ and n.

The notion of testing boolean functions in this framework goes back to the seminal work of Rubinfeld and Sudan [38], and has several connections to complexity theory (in particular PCPs and hardness of approximation), as well as computational learning theory [36]. Over the last two decades, researchers have exerted a considerable amount of effort in testing various properties of a function f, such as whether f is a linear function [8], whether f is isomorphic to a given function [7], [15], [1], whether f is a k-junta [22], [4], [5], a monotone function [26], [23], a dictator [35], a halfspace [30], an s-sparse polynomial, a size-s decision tree, etc. [18] (see, e.g., the survey of [37]).

Over the course of this effort, a variety of techniques have been developed for designing property testing algorithms, thus proving testing upper bounds. However, as is often the case in theoretical computer science, lower bounds are harder to come by. Although several lower bounds for specific problems are known, few general techniques are known beyond the use of Yao's minimax lemma.

Communication complexity is one technique that has proven effective for proving lower bounds in other areas of computer science. In a typical setup, two parties, Alice and Bob, each have an input and they would like to decide something about their joint input. Their computational power is unbounded, but they would like to compute the answer with as little *communication* as possible.

The communication complexity framework has been well-studied, and in particular several problems are known to require a large amount of communication. These include SET-DISJOINTNESS, INDEX, INNER-PRODUCT, and GAP-HAMMING-DISTANCE. The hardness of these and related problems has been used to obtain lower bounds in many areas such as streaming algorithms, circuit complexity, data structures, and proof complexity [29], [28], [31].

Property testing and communication complexity have striking similarities. Both involve parties with unbounded computational power (in one case, the tester, and in the other case, the communicating players), and both involve algorithms which are restricted by the parties' limited access to their input. Despite these similarities, no previous connection between these fields has been made.

In this work we show that in fact there is a strong connection between testing and communication complexity. In particular, we show how to reduce certain communication problems to testing problems, thus showing that communica-

*This work was supported in part by the National Basic Research Program of China Grant 2007CB807900, 2007CB807901, and the National Natural Science Foundation of China Grant 61033001, 61061130540, 61073174.

1093-0159/11 $26.00 © 2011 IEEE
DOI 10.1109/CCC.2011.31

tion lower bounds imply lower bounds for property testing.

This represents a new approach to proving testing lower bounds. For a particular testing problem $\mathcal{P}$ that we would like to bound, instead of starting from "scratch" by studying the structure of $\mathcal{P}$, we seek a connection between $\mathcal{P}$ and a hard communication problem. If we can find such a connection, then we can reduce the work involved. As we will show, this approach turns out to be quite fruitful, both for proving new bounds, and for giving simpler proofs of known bounds.

A. Our Results

TESTING k-LINEAR FUNCTIONS. The boolean function $f : \{0,1\}^n \to \{0,1\}$ is *linear*, i.e. a parity function, when there is a set $S = \{i_1, \ldots, i_s\} \subseteq [n]$ such that for every $x \in \{0,1\}^n$, $f(x) = x_{i_1} \oplus \cdots \oplus x_{i_s}$. When $|S| = k$, we say that f is a *k-linear* function.

The problem of testing k-linear functions was first studied by Fischer et al. [22]. The best lower bound is due to Goldreich [25], who showed that $\Omega(\sqrt{k})$ queries are required to test k-linear functions. He also showed that non-adaptive testers require $\Omega(k)$ queries to test the same property, and conjectured that this stronger lower bound holds for all testers (adaptive or not).[1]

We confirm Goldreich's conjecture. As a result, we also obtain lower bounds on the query complexity for testing juntas, testing functions of low Fourier degree, and testing sparse polynomials:

Theorem I.1. *Fix $1 < k < n - 1$. Then $\Omega(\min\{k, n-k\})$ queries are required to test*

(i) k-linear functions,
(ii) k-juntas,
(iii) functions of Fourier degree at most k, and
(iv) functions with k-sparse polynomial representation in $\mathbb{F}_2$.

We define these properties formally and prove Theorem I.1 in Section III.

Remark 1. In parallel work, Daniel Kane and the first author simultaneously obtained a different proof of Goldreich's conjecture via Fourier-analytic methods [6].

Remark 2. Goldreich has observed that our technique can also be used to resolve two other conjectures from [25]. The first conjecture is related to testing whether a function is computable by a small-width branching program; this conjecture is proven and generalized using our approach in [11]. The second conjecture involves testing whether a function belongs to a certain subclass of linear functions over $GF(3)$; we discuss the details in the full version of this paper.

We note that Theorem I.1 also has implications for the problem of *isomorphism testing*, or testing whether an unknown function f is equivalent, up to permutation of variables, to a fixed function $g : \{0,1\}^n \to \{0,1\}$. Alon and Blais showed that for most functions g, testing g-isomorphism non-adaptively requires $\Omega(n)$ queries [1]. Similarly, Chakraborty et al. showed that for every $k \leq n$, there exists a k-junta g such that testing g-isomorphism requires $\Omega(k)$ queries [15]. Both of these results are non-constructive, and they raise the question of whether we can identify an *explicit* class of functions for which the same lower bounds apply. Theorem I.1 shows that the class of k-linear functions satisfies this requirement.

TESTING MONOTONICITY. Fix $R \subseteq \mathbb{R}$. The function $f : \{0,1\}^n \to R$ is *monotone* if for any two inputs $x, y \in \{0,1\}^n$ where $x_1 \leq y_1, \ldots, x_n \leq y_n$, we have that $f(x) \leq f(y)$. The problem of testing monotonicity was first studied by Goldreich et al. [26], who introduced a natural tester: sample random edges from the hypercube and verify that the function is monotone on those edges. For ranges of size $|R|$, this algorithm requires $O(n \log |R|)$ queries [19]; an important open problem in property testing is to determine whether there exist more efficient monotonicity testers.

Despite much attention to monotonicity testing [2], [20], [26], [19], [23], [3], [9], lower bounds for the query complexity of this problem have been elusive. Previously, the best bound for non-adaptive testers was only $\Omega(\log n)$ [23] – this translates to a $\Omega(\log \log n)$ lower bound for general (adaptive) testers.[2] We provide a significant improvement to this lower bound for functions with large ranges:

Theorem I.2. *Testing $f : \{0,1\}^n \to R$ for monotonicity requires $\Omega(\min\{n, |R|^2\})$ queries.*

Notably, Theorem I.2 gives the first progress on the natural-monotonicity-tester problem mentioned above: it shows that for $\sqrt{n} \leq |R| \leq \text{poly}(n)$, no monotonicity tester can improve on the query complexity of the natural tester by more than a logarithmic factor. We note, however, that this problem is still open in the important special case when $R = \{0,1\}$.

By a recent result of Seshadhri and Vondrak [39], Theorem I.2 also gives a new lower bound for the query complexity of testing submodularity; see Section IV for details.

TESTING CONCISE REPRESENTATIONS. Parnas, Ron, and Samorodnitsky [35] showed that testing whether a function can be represented by a monotone DNF with at most s terms

[1] We note that Goldreich's conjecture and the results in [25] are stated in terms of testing $\leq k$-linear functions (the class of functions that are parities on at most k bits), but it is easy to see that the proofs in [25] give identical lower bounds for testing k-linearity. It is also easy to see that our lower bounds for testing k-linearity give identical bounds for testing $\leq k$-linearity.

[2] Stronger bounds have been established for testers with one-sided error – see [23], [9] for details.

Class of functions	Our bound	Previous lower bounds	Upper bounds
k-linear	$\boldsymbol{\Omega(k)}$	$\Omega(\sqrt{k})$ [25] $\Omega(k)$ (n.a.) [25]	$O(k \log k)$ [15] $O(n)$ (trivial)
k-juntas	$\Omega(k)$	$\Omega(k)$ [16]	$O(k \log k)$ [5]
Fourier degree $\leq d$	$\Omega(d)$	$\Omega(d)$ [15]	$2^{O(d)}$ [18], [15]
s-sparse $GF(2)$-polynomials	$\boldsymbol{\Omega(s)}$	$\Omega(\sqrt{s})$ [14]	$\tilde{O}(s)$ [14]
monotone $f : \{0,1\}^n \to R$	$\boldsymbol{\Omega(\min\{n, \lvert R\rvert^2\})}$	$\Omega(\log n)$ (n.a.) [23] $\Omega(n)$ (n.a., 1-s.) [9]	$O(n \log \lvert R\rvert)$ [19]
submodular $f : \{0,1\}^n \to \mathbb{R}$	$\boldsymbol{\Omega(n)}$	$\Omega(\log n)$ (n.a.) [23], [39] $\Omega(n)$ (n.a., 1-s.) [9], [39]	$2^{O(\sqrt{n} \log n)}$ [39]
size-s branching programs, size-s boolean formulas	$\Omega(\log s)$	$s^{\Omega(1)}$ [14]	$\tilde{O}(s)$ [14]
s-term DNF formulas	$\Omega(\log s)$	$\Omega(\log s)$ [14]	$\tilde{O}(s)$ [14]
size-s decision trees	$\boldsymbol{\Omega(s)}$ **(1-s.)**	$\Omega(\log s)$ [14]	$\tilde{O}(s)$ [14]
signed k-majority	$\boldsymbol{\Omega(k/\log k)}$ **(1-s.)** for $k \leq \gamma n$, $\gamma \in (0,1)$	$\Omega(k^{1/12})$(n.a.)[3] [7], [30] for $k \leq \frac{3}{4}n$	$O(\sqrt{n})$ [30] for $k = n$

Table I
OUR RESULTS. Bold font indicates an improvement over the previous bounds. Bounds labeled with (n.a.) apply only to non-adaptive testers; bounds marked with (1-s.) only apply to testers with one-sided error. All other bounds apply to adaptive testers with two-sided error.

can be done with a number of queries that depends only on s. This result was generalized by Diakonikolas et al. [18], who introduced the method of *testing by implicit learning* and showed that this method can be used to test whether a function can be represented by a DNF with few terms, by a small decision tree, by a small boolean formula, etc.

Our technique gives lower bounds on the query complexity for a number of these properties:

Theorem I.3. *At least $\Omega(\log s)$ queries are required to test*
(i) size-s decision trees,
(ii) size-s branching programs,
(iii) s-term DNFs, and
(iv) size-s boolean formulas.

Remark. In simultaneous and independent work, Chakraborty et al. prove matching $\Omega(\log s)$ bounds for s-term DNFs and size-s decision trees, and stronger $\mathrm{poly}(s)$ lower bounds for size-s boolean formulas and size-s branching programs [14].

The proof of Theorem I.3 can also be extended to answer a question of Fischer et al. [22]: they asked if the query complexity of testing k-juntas can be reduced if the tester is only required to reject functions that are far from $(k+t)$-juntas for some $t > 0$. We show that the answer to this question is "no" for any $t \leq O(\sqrt{k})$:

Theorem I.4. *Fix $k \leq \frac{3}{4}n$ and $t > 0$. Any algorithm that accepts k-juntas and rejects functions $\frac{1}{4}$-far from $(k+t)$-juntas with high probability must make $\Omega\big(\min\{(\frac{k}{t})^2, k\} - \log k\big)$ queries.*

We prove Theorems I.3 and I.4 in Section V.

TESTERS WITH ONE-SIDED ERROR. The technique we introduce for proving new lower bounds can also be used to prove lower bounds for testers with one-sided error (that is, testers which accept functions with probability 1 if they have property $\mathcal{P}$, and reject them with probability at least $2/3$ if they are far from having property $\mathcal{P}$). As a first application, we get a much stronger lower bound for the query complexity of testing decision trees with one-sided error:

Theorem I.5. *At least $\Omega(s)$ queries are required to test size-s decision trees with one-sided error.*

We also obtain a lower bound on the query complexity of one-sided testers for a subclass of halfspaces, the class of "signed" majority functions on k variables.

Theorem I.6. *Fix any constant $\gamma \in (0,1)$. For $k \leq \gamma n$, at least $\Omega(k/\log k)$ queries are required to test signed k-majorities with one-sided error.*

See Section VI for more information about the history of these problems and the proofs of Theorems I.5 and I.6.

B. Techniques

The main idea behind all of our bounds is to set up a communication game, where Alice has a function f, Bob has a function g, and they want to determine whether a

[3]The lower bound stated here is not found explicitly in [7], but can be obtained using the arguments in that paper.

joint function h, which is some combination of f and g (usually the XOR), has a particular property. We can then relate the number of queries required to test whether h has this property to the number of bits Alice and Bob need to communicate.

This technique is best illustrated by example. In fact, we can give a very simple sketch of Theorem I.1, by showing how to reduce a version of the well-known SET-DISJOINTNESS problem to testing k-linearity. Suppose Alice and Bob both have sets of size k from a universe of size n. Suppose further that their sets are guaranteed to either intersect in one place, or not at all, and they want to decide which is the case. It is well-known that the communication complexity of this problem is $\Omega(k)$ [27].

One way Alice and Bob can solve this set intersection problem is by forming linear functions based on their two sets. Alice forms the function $f = \chi_A$ and Bob forms the function $g = \chi_B$, where χ_A and χ_B are both k-linear functions. It is easy to see that the joint function $h = f \oplus g$ is $2k$-linear if the sets don't intersect, and $(2k-2)$-linear if they do. Note that every $(2k-2)$-linear function is $1/2$-far from being $2k$-linear (see Fact III.1). Therefore, they can determine if their sets intersect by each running a *testing* algorithm for $2k$-linearity on h. Whenever Alice's tester queries $h(x)$, she asks Bob for $g(x)$, and whenever Bob's tester queries $h(x)$, he asks Alice for $f(x)$ (we assume Alice and Bob use shared, public randomness to determine which queries to make, so exchanging x is unnecessary). The total number of bits communicated is then twice the number of queries of the tester. Since we can lower bound the number of bits communicated by $\Omega(k)$, this implies that testing $2k$-linearity also requires $\Omega(k)$ queries. By scaling k, we achieve the first part of Theorem I.1.

II. From Communication Complexity to Property Testing

In this section, we formalize the notions of query complexity for property testers, and of communication complexity.

PROPERTY TESTING. The query complexity $Q(\mathcal{P})$ of property $\mathcal{P}$ is the minimum cost of an adaptive tester for $\mathcal{P}$ with two-sided error. $Q^1(\mathcal{P})$ is the cost of the best algorithm that tests $\mathcal{P}$ with one-sided error. $Q^{\mathrm{na}}(\mathcal{P})$ is the query complexity of non-adaptive testers for $\mathcal{P}$.[4]

COMMUNICATION COMPLEXITY. We are primarily interested in (public coin) randomized protocols with one-sided and two-sided error. Let $R_\epsilon(f)$ denote the minimum cost of a randomized protocol that computes f with probability $\geq 1-\epsilon$. For $z \in \{0,1\}$, $R^z_\epsilon(f)$ denotes the cost of the best protocol that correctly outputs f whenever $f(x,y) \neq z$ and outputs $f(x,y)$ with probability $\geq 1-\epsilon$ whenever $f(x,y) = z$. Similarly, we let $R^{\rightarrow}_\epsilon(f)$ and $R^{\rightarrow,z}_\epsilon(f)$ denote the randomized communication complexity of one-way protocols (with one-sided error). Unless otherwise specified, we fix $\epsilon := 1/3$ and drop the subscript.

It might seem counterintuitive to define $R^z(f)$ as the cost of the best protocol that is always correct when $f(x,y) \neq z$; it is defined in this way because of its connection to nondeterministic communication complexity. Specifically, let $C^z(f)$ denote the minimum number of monochromatic rectangles needed to cover the z-inputs of f, and define $N^z(f) := \log C^z(f)$. Then, we have

Fact II.1 ([29] Proposition 3.7). *For all constant $0 < \epsilon < 1$, $N^z(f) \leq R^z_\epsilon(f) + O(\log n)$.*

For more details, see the standard text by Kushilevitz and Nisan [29].[5] It is worth noting that in [29], the definitions for the different notions of communication complexity are defined in terms of total functions f, whereas we are primarily concerned with partial functions. However, the definitions generalize, and it is easy to verify that Fact II.1 also applies to partial functions.

Given a property $\mathcal{P}$, functions f, g, and a "combining function" $h = h(f,g)$, we define the following communication game $\mathrm{C}_{h,\mathcal{P}}$: Alice and Bob receive f and g respectively, and they want to decide if h has property $\mathcal{P}$ or is ϵ-far from all functions that have $\mathcal{P}$. For most of our applications, f and g will be boolean functions, and we will define $h := f \oplus g$; however, this need not always be the case. When we use more exotic definitions of h, we note so explicitly. The following lemma formalizes the connection between property testing and communication complexity.

Lemma II.2. *For any function h and any property $\mathcal{P}$ for h,*

1) $R(\mathrm{C}_{h,\mathcal{P}}) \leq 2\,Q(\mathcal{P})$,
2) $R^1(\mathrm{C}_{h,\mathcal{P}}) \leq 2\,Q^1(\mathcal{P})$, *and*
3) $R^{\rightarrow}(\mathrm{C}_{h,\mathcal{P}}) \leq Q^{\mathrm{na}}(\mathcal{P})$.

Remark. Lemma II.2 assumes that f and g have boolean range. In the more general case where the range of f and g has cardinality r, the bounds on the right-hand side must be multiplied by an extra factor of $\log r$.

Proof: Given a t-query general testing algorithm for $\mathcal{P}$, we create a protocol for $\mathrm{C}_{h,\mathcal{P}}$ in the following manner. Alice and Bob use public randomness to adaptively generate queries. For each query x, Alice and Bob exchange $f(x)$ and $g(x)$, enabling each player to compute $h(x)$. After t queries (and $2t$ bits of communication), both players use the testing algorithm to determine if h has $\mathcal{P}$.

[4]Typically, the query complexity of a testing algorithm depends on the distance parameter ϵ. Throughout this work, we will assume ϵ is any small, fixed constant (say $\epsilon = 0.01$), and for simplicity we will state all query complexity bounds only in terms of the other parameters involved.

[5]The relation between randomized and nondeterministic communication complexity is actually for *private coin* protocols; however by Newman's Theorem [32], the public-coin and private-coin complexities essentially differ by at most a $O(\log n)$ term.

The proof connecting one-sided property testing to protocols with one-sided error is analogous. In the non-adaptive case, we construct the following one-way protocol: Alice and Bob generate queries $x_1, \ldots, x_t$ in advance. Alice sends Bob a single t-bit message, consisting of $\{f(x_i) : i \in [t]\}$. Bob then computes $\{h(x_i)\}$ and outputs 1 if and only if the tester accepts h. ∎

A. Communication Complexity Problems

We achieve all of our testing lower bounds via Lemma II.2. To prove lower bounds for $\mathrm{C}_{h,\mathcal{P}}$, we reduce from one of several standard communication complexity problems. However, we often require special flavors of these problems—either we need protocols with one-sided error, or we require the input to be restricted in some *balanced* way. Let $n \in \mathbb{N}, t := t(n)$, and $x, y \in \{0,1\}^n$. We are interested in the following functions:

SET-DISJOINTNESS. Alice and Bob are given x and y and compute

$$\textsc{disj}(x,y) := \bigvee_{i=1}^{n} x_i \wedge y_i \,.$$

It is well-known that $R(\textsc{disj}) = \Omega(n)$. We use k-DISJ, a balanced version of DISJ with the promise that $|x| = |y| = k$ and that $x_i \wedge y_i = 1$ for at most one i. It is known that $R(k\text{-}\textsc{disj}) = \Omega(k)$ [27].

GAP-EQUALITY. Alice and Bob are given n-bit strings x and y respectively and wish to compute

$$\textsc{geq}_{n,t}(x,y) := \begin{cases} 1 & \text{if } x = y\,, \\ 0 & \text{if } \Delta(x,y) = t\,, \\ * & \text{otherwise.} \end{cases}$$

We drop the subscripts when n is clear from context and $t = n/8$. We are interested in $R^z(\textsc{geq})$. The standard public-coin EQUALITY protocol gives $R^0(\textsc{geq}) = O(1)$. For protocols that only err when $\textsc{geq}(x,y) = 1$, the complexity is drastically different.

Buhrman, Cleve, and Wigderson [12] proved an $\Omega(n)$ lower bound on the deterministic communication complexity of $\textsc{geq}_{n,n/2}$; their result extends to other gap sizes and to randomized protocols with one-sided error.

Lemma II.3 ([12]). *$R^1(\textsc{geq}_{n,t}) = \Omega(n)$ for all even $t = \Theta(n)$.*[6]

We include a proof for completeness. The proof of this lemma uses the following celebrated result of Frankl and Rödl.

[6]Curiously, the parity of t turns out to be necessary. Since $\Delta(x,y) = |x| + |y| - 2|x \cap y|$, Alice and Bob can deterministically distinguish $x = y$ from $\Delta(x,y)$ being odd in $O(\log n)$ bits by exchanging $|x|$ and $|y|$ and checking the parity of $|x| + |y|$. This does not affect our property testing lower bounds.

Fact II.4 ([24], Theorem 1.10). *For all constant $0 < \rho < 1/2$, there exists $\delta_\rho = \delta(\rho)$ such that for all even $d \in [\rho n, (1-\rho)n]$, if $S \subseteq \{0,1\}^n$ and $\Delta(x,y) \neq d$ for all $x, y \in S$, then $|S| \leq 2^{n(1-\delta_\rho)}$.*

Proof of Lemma II.3: Fix a 1-monochromatic rectangle R for $\textsc{geq}_{n,t}$. Let $T_R := \{x : (x,x) \in R\}$, and consider any $x, y \in T_R$. Since R is a rectangle, $(x,y) \in R$; as R is monochromatic, it follows that $\Delta(x,y) \neq t$ for all $x, y \in T_R$. By Fact II.4, $|T_R| \leq 2^{n(1-\delta_\rho)}$. Trivially, there are 2^n (x,y) pairs such that $x = y$; each 1-monochromatic rectangle contains at most $2^{n(1-\delta_\rho)}$ such pairs. Therefore, we have $C^1(\textsc{geq}_{n,t}) \geq 2^{n\delta_\rho} = 2^{\Omega(n)}$. The rest of the proof follows from Fact II.1. ∎

GAP-HAMMING-DISTANCE. Alice and Bob are given n-bit strings x and y respectively and wish to compute

$$\textsc{ghd}_{n,t}(x,y) := \begin{cases} 1 & \text{if } \Delta(x,y) \geq n/2 + t\,, \\ 0 & \text{if } \Delta(x,y) \leq n/2 - t\,, \\ * & \text{otherwise.} \end{cases}$$

The standard gap size for GHD is $t = \Theta(\sqrt{n})$; in this case, we drop the subscripts and use just GHD. A tight lower bound of $R(\textsc{ghd}) = \Omega(n)$ is known, due to Chakrabarti and Regev [13]. An easy padding argument (implicit in [10]) shows that $R(\textsc{ghd}_{n,t}) = \Omega((n/t)^2)$ for all $t = \Omega(\sqrt{n})$.

We consider an extended version of GHD. In $\textsc{eghd}_{n,k,t}$, Alice and Bob's inputs x, y are n-bit strings, with the promise that $|x| = |y| = k/2$, and they wish to distinguish $\Delta(x,y) \geq k/2 + t$ from $\Delta(x,y) \leq k/2 - t$.

Lemma II.5. *For all t and all $k \leq n$,*

$$R(\textsc{eghd}_{n,k,t}) = \Omega(\min\{(k/t)^2, k\} - \log k).$$

In particular, when $k = n$, we show that $\textsc{ghd}_{n,t}$ remains hard even when $|x| = |y| = n/2$.

In the proof of the lemma, let $\textsc{cost}(P)$ denote the maximum number of bits sent in a protocol P. We use $\circ$ to denote string concatenation and 0^k (1^k) to denote the string of k consecutive zeros (ones).

Proof: First, we prove the lemma for the case $k = n$ by reduction from GHD. Let P be the best protocol for $\textsc{eghd}_{n,k,t}$. Fix $m := n/4$, and let x, y denote two arbitrary inputs to $\textsc{ghd}_{m,t}$. Alice and Bob construct $4m$-bit inputs $\hat{x}, \hat{y}$ such that $|\hat{x}| = |\hat{y}| = 2m$ and that $\textsc{ghd}_{m,t}(x,y) = \textsc{eghd}_{n,k,t}(\hat{x}, \hat{y})$. Then, the protocol outputs $P(\hat{x}, \hat{y})$. Next we describe how to construct $\hat{x}$ and $\hat{y}$. Let z be the absolute value of $(|x| - |y|)$, and consider the following $2m$-bit strings.

$$x' := x \circ 1^{m-|x|} \circ 0^{|x|},$$
$$y' := y \circ 1^{m-|y|} \circ 0^{|y|}\,.$$

Note that $|x'| = |y'| = m$ and that

$$|\Delta(x', y') - (m/2 + z)| \geq t.$$

These strings are balanced, but in general, the Hamming distance is not centered around $2m \pm t$. To get balanced strings whose Hamming distance is centered, Alice and Bob again append their inputs, this time creating $4m$-bit strings $\hat{x}$ and $\hat{y}$ such that

$$\hat{x} := x' \circ 1^m \circ 0^m,$$
$$\hat{y} := y' \circ 1^{(m+2z)/4} \circ 0^m \circ 1^{(3m-2z)/4}.$$

It's easy to see that $\hat{x}$ and $\hat{y}$ are $4m$-bit strings with Hamming weight $2m$. Their Hamming distance increases by $(3m-2z)/2$, so $|\Delta(\hat{x},\hat{y}) - 2m| \geq t$.

In our protocol Q for $\text{GHD}_{m,t}$, Alice and Bob exchange $|x|$ and $|y|$, construct $\hat{x}, \hat{y}$, and output $P(\hat{x},\hat{y})$. By construction, it's easy to see that $\text{GHD}_{m,t}(x,y) = \text{EGHD}_{n,k,t}(\hat{x},\hat{y})$, hence Q is correct whenever P is correct. The cost of Q equals $\text{COST}(P) + 2\log m$. Therefore, we have

$$\text{COST}(P) = \text{COST}(Q) - 2\log m.$$

Hence, when $t = O(\sqrt{n})$ then

$$\text{COST}(P) = \Omega(m) - 2\log m = \Omega(n),$$

and when $t = \Omega(\sqrt{n})$ then

$$\text{COST}(P) = \Omega((m/t)^2) - 2\log m = \Omega((n/t)^2 - 2\log n).$$

Proving the general case occurs by a simple padding argument. Specifically, take inputs to $\text{EGHD}_{k,k,t}$ and extend them to n bit strings by appending with 0^{n-k}. ■

III. Testing k-Linearity and Related Properties

In this section we prove Theorem I.1. Recall that a k-linear function is a function of the form $f(x) = \sum_{i \in S} x_i \pmod 2$ for some set $S \subseteq [n]$ where $|S| = k$. The definitions of the other properties in the statement of Theorem I.1 are as follows:

Definition (Junta)**.** The function $f : \{0,1\}^n \to \{0,1\}$ is a k-junta if there is a set $J \subseteq [n]$ of size $|J| \leq k$ such that for every $x, y \in \{0,1\}^n$ where $x_i = y_i$ for each $i \in J$, $f(x) = f(y)$.

Definition (Low Fourier degree)**.** For convenience when discussing Fourier degree we will represent boolean functions using range $\{-1,1\}$ instead of $\{0,1\}$. It is well known that every boolean function $f : \{0,1\}^n \to \{-1,1\}$ has a unique representation of the form $f(x) = \sum_{S \subseteq [n]} \hat{f}(S)\chi_S(x)$, where $\chi_S = (-1)^{\sum_{i \in S} x_i}$ and $\hat{f}(S) \in \mathbb{R}$. The terms $\hat{f}(S)$ are the *Fourier coefficients* of f, and the *Fourier degree* of f is the maximum value of $k \geq 0$ such that $\hat{f}(S) \neq 0$ for some set S of size $|S| = k$.[7]

Definition (Sparse polynomials)**.** Every boolean function $f : \{0,1\}^n \to \{0,1\}$ also has a unique representation as a polynomial over $\mathbb{F}_2$. We say that f is a *k-sparse polynomial* if its representation over $\mathbb{F}_2$ has at most k terms.

The following facts about k-linear functions will be used in the proof of Theorem I.1:

Fact III.1. *A $(k+2)$-linear function is $\frac{1}{2}$-far from (i) k-linear functions, from (ii) k-juntas, from (iii) functions of Fourier degree at most k, and $\frac{1}{20}$-far from (iv) k-sparse polynomials.*

Proof: We first prove part (iii). Parts (i) and (ii) will follow immediately from the observation that k-juntas and k-linear functions are subclasses of functions with Fourier degree at most k.

Let f be a $(k+2)$-linear function over the variables of some set $T \subseteq [n]$ where $|T| = k+2$, and let g be any function of Fourier degree at most k. For convenience, we will represent f and g as functions from $\{0,1\}^n$ to $\{-1,1\}$. Since f is a linear function over the variables in T, we know that $\hat{f}(T) = 1$, and $\hat{f}(S) = 0$ for all $S \neq T$. Moreover, since g has Fourier degree k and $|T| > k$, we know by definition that $\hat{g}(T) = 0$. Thus by Parseval's theorem

$$\mathop{\mathrm{E}}_{x}[f(x)g(x)] = \sum_{S \subseteq [n]} \hat{f}(S)\hat{g}(S) = 0$$

which implies $\Pr_x[f(x) \neq g(x)] = 1/2$.

Finally, part (iv) is a special case of a more general theorem of Diakonikolas et al. [18, Thm. 36]. ■

Theorem I.1 (Restated)**.** *Fix $1 < k < n-1$. Then $\Omega(\min\{k, n-k\})$ queries are required to test (i) k-juntas, (ii) k-linear functions, (iii) functions of Fourier degree at most k, and (iv) functions with k-sparse polynomial representation in $\mathbb{F}_2$.*

Proof: We will first prove the theorem for k in the range $k \in (1, n/2)$, then discuss how to handle other values of k.

Let k be even and define $k' = \frac{k}{2} + 1$. We will show a reduction from the k'-DISJ problem. An instance of this problem is a pair of sets $A, B \subseteq [n]$ such that $|A| = |B| = k'$ and $|A \cap B| \in \{0,1\}$. Alice and Bob each receive one of the sets and they must determine whether $|A \cap B| = 0$. As we saw in Section II-A, $R(k'\text{-DISJ}) \geq \Omega(k)$.

Here is a protocol to solve the k'-DISJ problem: Alice and Bob start by building the boolean functions $\text{Parity}_A, \text{Parity}_B : \{0,1\}^n \to \{0,1\}$ that return the parity of the bits in A and B, respectively. They then communicate to determine if $h := \text{Parity}_A \oplus \text{Parity}_B$ is a k-linear function or a $(k+2)$-linear function. Since $\text{Parity}_A \oplus \text{Parity}_B = \text{Parity}_{A \triangle B}$, h is a k-linear function iff $|A \cap B| = 1$.

Define $C_{h,\mathcal{P}}$ to be the communication game where Alice and Bob each receive a function – call these functions f and g – with the promise that $f \oplus g$ is a linear function on exactly k or $k+2$ bits and they must accept iff $f \oplus g$ is a k-linear function. The above reduction shows that $R(C_{h,\mathcal{P}}) \geq R(k\text{-DISJ}) \geq \Omega(k)$. By Lemma II.2, any

[7] For more details on the Fourier representation of boolean functions see, e.g., [17], [33].

testing algorithm that distinguishes k-linear and $(k+2)$-linear functions with probability at least $2/3$ must make at least $\Omega(k)$ queries. The theorem then follows from the observation that k-linear functions satisfy properties (i)–(iv) while Fact III.1 shows that $(k+2)$-linear functions are far from those same properties.

To handle k in the range $k \in (n/2, n-1)$, note that the query complexity of distinguishing whether a function is k-linear versus $(k+2)$-linear is equivalent to the query complexity of distinguishing whether a function is $(n-k)$-linear versus $(n-k-2)$-linear. This is because we can replace the function h being tested by $h \oplus \chi_n$. Thus for $k \in (n/2, n-1)$, the complexity of testing any of these properties is $\Omega(n-k)$.

For the special case when $k = n/2$, we can show $\Omega(n)$ queries are required via a simple padding argument. We reduce the $k = 3n/4$ case (say) to the $k = n/2$ case by using the function h constructed by Alice and Bob to construct a padded h' over a larger space- i.e. h' has the form $h : \{0,1\}^{n'} \to \{0,1\}$ where $n' = 3n/2$ and h' just applies h to the first n variables. Thus a distinguisher for whether h' is $\frac{n'}{2}$-linear versus $(\frac{n'}{2}+2)$-linear would clearly yield a distinguisher for whether h is $\frac{3n}{4}$-linear versus $(\frac{3n}{4}+2)$-linear. ∎

IV. Testing Monotonicity and Submodularity

Theorem I.2 (Restated)**.** *Testing $f : \{0,1\}^n \to R$ for monotonicity requires $\Omega(\min\{n, |R|^2\})$ queries.*

Proof: We prove the theorem in three steps. First, we give an $\Omega(n)$ lower bound for the case when $R = \mathbb{Z}$. Secondly, we handle the case where $|R| = \sqrt{n}$ by a standard range reduction argument. Finally, we give an $\Omega(|R|^2)$ bound for small $|R|$ by reducing from the $|R| = \sqrt{n}$ case.

Suppose $R = \mathbb{Z}$. Then, we apply a reduction from the DISJ problem. Let $A, B \subseteq [n]$ be the subsets received by Alice and Bob, respectively. Alice and Bob can determine whether A and B are disjoint with the following protocol: Alice builds the function $\chi_A : \{0,1\}^n \to \{-1,1\}$ defined by $\chi_A(x) = (-1)^{\sum_{i \in A} x_i}$. Similarly, Bob constructs the function $\chi_B : \{0,1\}^n \to \{-1,1\}$. They then communicate to test whether the function $h : \{0,1\}^n \to \mathbb{R}$ defined by $h(x) = 2 \cdot |x| + \chi_A(x) + \chi_B(x)$ is monotone or whether it is $1/8$-far from monotone.

To establish the correctness of the protocol, we need to establish two facts: (1) when A and B are disjoint, the function h is monotone, and (2) when A and B are not disjoint, h is $1/8$-far from monotone.

Fix $i \in [n]$. For $x \in \{0,1\}^n$, let $x_0, x_1 \in \{0,1\}^n$ be the vectors obtained by fixing the ith coordinate of x to 0 and to 1, respectively. For any set $S \subseteq [n]$,

$$\chi_S(x_1) = (-1)^{\mathbf{1}[i \in S]} \cdot \chi_S(x_0).$$

Therefore, when $i \notin A$ and $i \notin B$,

$$h(x_1) - h(x_0) = 2\,|x_1| - 2\,|x_0| = 2 > 0;$$

when $i \in A$ and $i \notin B$,

$$h(x_1) - h(x_0) = 2\,|x_1| - 2\,|x_0| - 2\,\chi_A(x_0) \geq 0;$$

and similarly when $i \notin A$ and $i \in B$,

$$h(x_1) - h(x_0) = 2\,|x_1| - 2\,|x_0| - 2\,\chi_B(x_0) \geq 0.$$

So when $i \notin A \cap B$, the function h is monotone on each edge (x_0, x_1) in the ith direction. As a result, when A and B are disjoint the function h is monotone. This completes the proof of fact (1).

Consider now the case where $A \cap B \neq \emptyset$. When $i \in A \cap B$,

$$h(x_1) - h(x_0) = 2\,|x_1| - 2\,|x_0| - 2\,\chi_A(x_0) - 2\,\chi_B(x_1).$$

This implies that for each x where $\chi_A(x_0) = \chi_B(x_0) = 1$, $h(x_1) < h(x_0)$. Partition $\{0,1\}^n$ into 2^{n-1} pairs that form the endpoints to all the edges in the ith direction. Exactly $\frac{1}{4}$ of these pairs will satisfy the condition $\chi_A(x_0) = \chi_B(x_0) = 1$, and for each of these pairs, either $h(x_0)$ or $h(x_1)$ must be modified to make h monotone. Therefore, when A and B are not disjoint, then h is $\frac{1}{8}$-far from monotone and this completes the proof of fact (2).

To complete the proof in the case of $R = \mathbb{Z}$, define $C_{h,\text{Mon}}$ to be the communication game where Alice and Bob receive two functions f and g, and they must test whether the function h defined by $h(x) = 2 \cdot |x| + f(x) + g(x)$ is monotone or whether it is $1/8$-far from monotone. The argument above shows that $R(C_{h,\text{Mon}}) \geq R(\text{DISJ}) \geq \Omega(n)$. The lower bound thus follows from Lemma II.2.

To handle the case where $|R| = \sqrt{n}$, we sketch the proof of a standard range reduction argument (see, e.g., [9].) Specifically, we can assume without loss of generality that $R = \{-\frac{\sqrt{n}}{2}, \ldots, \frac{\sqrt{n}}{2}\}$ and we modify the construction of the function h to create h'

$$h'(x) = \begin{cases} -\frac{\sqrt{n}}{2} & \text{when } |x| - \frac{n}{2} < -\frac{\sqrt{n}}{2} + 1, \\ \frac{\sqrt{n}}{2} & \text{when } |x| - \frac{n}{2} > \frac{\sqrt{n}}{2} - 1, \\ |x| - \frac{n}{2} + \frac{\chi_A(x) + \chi_B(x)}{2} & \\ & \text{when } \big|\,|x| - \frac{n}{2}\big| \leq \frac{\sqrt{n}}{2} - 1. \end{cases}$$

It is easy to see that h' is identical to $h/2$, except when $\big|\,|x| - \frac{n}{2}\big| \geq \frac{\sqrt{n}}{2}$, which only occurs for a constant fraction of x's. Using the same reasoning as before, h' is monotone when A and B are disjoint, and a constant distance from monotone when A and B intersect. We leave the details to the reader.

Finally, suppose that $|R| = o(\sqrt{n})$, and let $m := |R|^2$. We'll use a q-query testing algorithm for f to create a q-query testing algorithm for functions $g : \{0,1\}^m \to \{0,1\}$.

Specifically, given g, create $h : \{0,1\}^n \to R$ by defining $h(x,y) := g(x)$ for $x \in \{0,1\}^m$ and $y \in \{0,1\}^{n-m}$. Clearly, if g is monotone then so is h. We now want to argue that if g is ϵ-far from monotone, then so is h. We do so by proving the contrapositive. Suppose that h is *not* ϵ-far from monotone. Let $\tilde{h}$ be the monotone function closest to h; thus, $\Pr_{x,y}[h(x,y) \neq \tilde{h}(x,y)] \leq \epsilon$. By an averaging argument, there exists y such that $\Pr_x[h(x,y) \neq \tilde{h}(x,y)] \leq \epsilon$. Define $\tilde{g} : \{0,1\}^m \to R$ as $\tilde{g}(x) := \tilde{h}(x,y)$. It's easy to see that $\Pr_x[g(x) \neq \tilde{g}(x)] = \Pr_{x,y}[h(x,y) \neq \tilde{h}(x,y)] \leq \epsilon$. Therefore, g is not ϵ-far from monotone.

Our testing algorithm for g is simple: test h and return the result. By the above claim, a correct answer for testing h gives a correct answer for testing g. Since testing g for monotonicity requires $\Omega(m) = \Omega(|R|^2)$ queries, the same bound holds for testing h. ∎

TESTING SUBMODULARITY. The real-valued function $f : \{0,1\}^n \to \mathbb{R}$ is *submodular* if for every $x, y \in \{0,1\}^n$, $f(x \vee y) + f(x \wedge y) \geq f(x) + f(y)$, where $(x \vee y)_i = \max\{x_i, y_i\}$ and $(x \wedge y)_i = \min\{x_i, y_i\}$.

Testing submodularity was first studied by Parnas, Ron, and Rubinfeld [34] for functions in low dimensions. Recently, Seshadhri and Vondrak [39] initiated the study of submodularity testing for functions over the boolean hypercube. They show that testing submodularity is at least as difficult as testing monotonicity (see Lemma 51 of [39]), and thus the monotonicity lower bound of Fischer et al. [23] implies a weak lower bound of $\Omega(\log \log n)$ for testing submodularity. Applying our Theorem I.2 instead, we get a much stronger lower bound:

Corollary IV.1. *Testing $f : \{0,1\}^n \to \mathbb{R}$ for submodularity requires $\Omega(n)$ queries.*

V. TESTING CONCISE REPRESENTATIONS

The following lemma regarding juntas is an important ingredient of the proof of Theorem I.3:

Lemma V.1 (Diakonikolas et al. [18])**.** *Let $\mathcal{P}$ be the class of all size-s decision trees, size-s branching programs, s-term DNFs, or size-s boolean formulas. Then every $(\log s)$-junta is in $\mathcal{P}$, while a random $(\log s + \log\log s)$-junta is 0.001-far from $\mathcal{P}$ with probability $1 - o(1)$.*

Theorem I.3 (Restated)**.** *At least $\Omega(\log s)$ queries are required to test (i) size-s decision trees, (ii) size-s branching programs, (iii) s-term DNFs, and (iv) size-s boolean formulas.*

Proof: Fix $\mathcal{P}$ to be the property consisting of all size-s decision trees, size-s branching programs, s-term DNFs, or size-s boolean formulas. Define $k := \log s$.

We prove that $\Omega(k)$ queries are required to test $\mathcal{P}$ with a reduction from the $\text{EGHD}_{n,4k/3,2\log k}$ problem. We can formulate the problem as follows: Alice and Bob receive $A, B \subseteq [n]$, respectively. Both sets have size $|A| = |B| = \frac{2}{3}k$. Alice and Bob must distinguish between the case where $|A \triangle B| \geq \frac{2}{3}k + 2\log k$ and the case where $|A \triangle B| \leq \frac{2}{3}k - 2\log k$. As we saw in Section II-A, $R(\text{EGHD}_{n,4k/3,2\log k}) \geq \Omega(k) = \Omega(\log s)$.

Alice and Bob can solve the EGHD problem with the following protocol: Alice generates a random $\frac{2}{3}k$-junta $f : \{0,1\}^n \to \{0,1\}$ whose relevant variables are identified by A. Similarly, Bob generates a random $\frac{2}{3}k$-junta $g : \{0,1\}^n \to \{0,1\}$ whose relevant variables are identified by B. Alice and Bob then test whether the function $f \oplus g$ is in $\mathcal{P}$ or is far from $\mathcal{P}$.

To see why the protocol correctly solves the $\text{EGHD}_{n,4k/3,2\log k}$ problem, we observe that the function $h = f \oplus g$ is a random junta on the set $A \cup B$ of variables. Since $|A| = |B| = \frac{2}{3}k$, then $|B \setminus A| = \frac{1}{2}|A \triangle B|$ and $|A \cup B| = |A| + |B \setminus A| = \frac{2}{3}k + \frac{1}{2}|A \triangle B|$. So when $|A \triangle B| < \frac{2}{3}k - 2\log k$, then h is a k-junta.[8] And when $|A \triangle B| > \frac{2}{3}k + 2\log k$, then h is a random $(k + \log k)$-junta. The correctness of the protocol follows from Lemma V.1.

We now complete the proof of the lower bound as we did in Theorems I.1 and I.2: define $C_{h,\text{Jun}}$ to be the communication game where Alice and Bob receive the functions $f, g : \{0,1\}^n \to \{0,1\}$ and must test whether $f \oplus g$ is in $\mathcal{P}$ or far from $\mathcal{P}$. The protocol above shows that $R(C_{h,\text{Jun}}) \geq R(\text{EGHD}_{n,4k/3,2\log k}) \geq \Omega(\log s)$. The Theorem follows from Lemma II.2. ∎

TESTING JUNTAS. Fischer et al. [22] asked if it is easier to test k-juntas if we are only required to reject functions that are far from $(k+t)$-juntas for some $t > 0$. The lower bound of Chockler and Gutfreund [16] gives a lower bound of $\Omega(k/t)$ queries for this task. (See also [18, App. E].) This bound is not sufficiently strong to answer Fischer et al.'s question for any $t \geq \omega(1)$.

Our proof of Theorem I.3, on the other hand, can easily be extended to show that for any $t \leq O(\sqrt{k})$, the task of distinguishing k-juntas from functions that are far from $(k+t)$-juntas requires (asymptotically) as many queries as the standard k-junta testing problem:

Theorem I.4 (Restated)**.** *Fix $k \leq \frac{3}{4}n$ and $t > 0$. Any algorithm that accepts k-juntas and rejects functions $\frac{1}{4}$-far from $(k+t)$-juntas with high probability must make $\Omega\big(\min\{(\frac{k}{t})^2, k\} - \log k\big)$ queries.*

Proof: We again define a reduction from the $\text{EGHD}_{n,4k/3,t}$ problem. As in the proof of Theorem I.3, Alice and Bob can solve their instance of the problem by building random juntas $f, g : \{0,1\}^n \to \{0,1\}$ on the sets $A, B \subseteq [n]$ of size $|A| = |B| = \frac{2}{3}k$ that they received. When $|A \triangle B| \leq \frac{2}{3}k - t$, then $f \oplus g$ is a k-junta, and when $|A \triangle B| \geq \frac{2}{3}k + t$ then $f \oplus g$ is a random $(k+t)$-junta. A

[8] In fact, h is a $(k - \log k)$-junta, but it is sufficient for our purposes to note that h is a k-junta.

random $(k+t)$-junta is $\frac{1}{4}$-far from $(k+t-1)$-juntas with probability $1-o(1)$, so this reduction and Lemma II.2 show that the relaxed version of junta testing is at least as hard as $\text{EGHD}_{n,4k/3,t}$. ∎

VI. Testers with One-Sided Error

TESTING DECISION TREES. We saw in Theorem I.3 that $\Omega(\log s)$ queries are required to test whether a function can be represented as a boolean decision tree with at most s nodes; for testers with one-sided error, we get an exponentially larger bound:

Theorem I.5 (Restated)**.** *At least $\Omega(s)$ queries are required to test size-s decision trees with one-sided error.*

Proof: We do a reduction from the GAP-EQUALITY problem. Assume that $s = 2^{n-1}$. Alice receives the string $a \in \{0,1\}^s$ and Bob receives $b \in \{0,1\}^s$. They must determine if $a = b$ or whether $\Delta(a,b) = \frac{s}{8}$.

Alice and Bob can solve their instance of the GEQ problem with the following protocol. Let the set of vectors $x \in \{0,1\}^n$ with even parity $\text{Parity}(x) = x_1 \oplus \cdots \oplus x_n = 0$ define an indexing of the bits of a. (I.e., fix a bijection between those strings and $[s]$.) Alice and Bob build the functions $f, g : \{0,1\}^n \to \{0,1\}$ by setting

$$f(x) = \begin{cases} a_x & \text{when } \text{Parity}(x) = 0, \\ 0 & \text{when } \text{Parity}(x) = 1, \end{cases}$$

and

$$g(x) = \begin{cases} b_x & \text{when } \text{Parity}(x) = 0, \\ 1 & \text{when } \text{Parity}(x) = 1. \end{cases}$$

Alice and Bob then test whether $f \oplus g$ can be represented with a decision tree of size at most $\frac{15}{16}2^n$; when it can, they answer $\Delta(a,b) = \frac{s}{8}$.

Let us verify the correctness of this protocol. For any $x \in \{0,1\}^n$ where $\text{Parity}(x) = 0$, we have that $(f \oplus g)(x) = a_x \oplus b_x$. Furthermore, for each x where $\text{Parity}(x) = 1$, we get $(f \oplus g)(x) = 1$. So when $a = b$, then $f \oplus g$ is the Parity function. This function requires a tree of size $2^n - 1$ to compute exactly, and is $\frac{1}{16}$-far from every decision tree of size at most $\frac{15}{16}2^n$. When $\Delta(a,b) = \frac{s}{8}$, consider the (complete) tree that computes $f \oplus g$ by querying x_i in every node at level i. This tree has $2^n - 1$ nodes, but for every input x where $a_x \neq b_x$, we have that the corresponding leaf has the same value as its sibling. So for each such input, we can eliminate one node in the nth level of the tree. Therefore, we can compute $f \oplus g$ with a decision tree of size at most $2^n - 1 - 2^{n-1}/8 < \frac{15}{16}2^n$.

To complete the proof, we introduce the communication game $C_{\oplus,\text{DT}}$ where Alice and Bob each receive a boolean function and they must determine if the sum of their functions can be represented with a decision tree of size $\frac{15}{16}2^n = \frac{15}{32}s$. The above reduction shows that $R^1(C_{\oplus,\text{DT}}) \geq R^1(\text{GEQ}) \geq \Omega(s)$. Lemma II.2 then implies the lower bound for testing size-s decision trees with one-sided error. ∎

TESTING SIGNED k-MAJORITIES. Our next bound is for testing whether a function $f : \{-1,1\}^n \to \{-1,1\}$ is a signed k-majority (for convenience, in this section we will switch notation and represent boolean values with ± 1 notation). A *signed majority* is a majority function with some variables negated, i.e. it is a halfspace of the form $f(x) = sgn(w \cdot x)$, where $w \in \{-1,1\}^n$. If $w \in \{-1,0,1\}^n$ and exactly k of the w_i's are non-zero, we say it is a *signed k-majority*.

Signed majorities were previously studied by Matulef et. al. [30], where they were referred to as $\{-1,1\}$*-weight halfspaces*. In that work, they show a non-adaptive lower bound of $\Omega(\log n)$ queries to test whether a function is a signed majority on all n variables. In [7], Blais and O'Donnell study the related problem of testing whether a function is a (non-signed) majority on exactly k out of n variables. When $k \leq \frac{3}{4}n$, they show a lower bound of $\Omega(k^{1/12})$ queries for non-adaptive algorithms with two-sided error.

We show that $\Omega(k/\log k)$ queries are required to test whether f is a signed k-majority with one-sided error. The argument in [7] can be adapted to show a non-adaptive, two-sided lower bound of $\Omega(k^{1/12})$ queries for this problem as well. Our bound is incomparable; it is asymptotically stronger and applies to adaptive algorithms, but only ones with one-sided error.

Theorem VI.1. *Fix any constant $\gamma \in (0,1)$. For $k \leq \gamma n$, at least $\Omega(k/\log k)$ queries are required to test signed k-majorities with one-sided error.*

Proof: We will show a reduction from the GAP-EQUALITY problem.

For a fixed k, define $k' = k/\gamma$ and note $k' \leq n$. Suppose Alice and Bob each have strings of length k' denoted s_A and s_B, which are promised to either be equal, or have Hamming distance $n - k$. For convenience, we will think of these strings as vectors over $\{-1,1\}^{k'}$.

Alice and Bob will each generate functions that are linear forms. Alice generates $f : \{-1,1\}^n \to \mathbb{R}$ by defining $f(x) = x \cdot s_A$, and Bob generates $g : \{-1,1\}^n \to \mathbb{R}$ by taking $g(x) = x \cdot s_B$. (For example, if $s_A = \langle -1,-1,1 \rangle$ Alice generates the function $f(x) = -x_1 - x_2 + x_3$). They then analyze the joint function $h : \{-1,1\}^n \to \{-1,1\}$ defined as $h(x) = \text{sgn}(\frac{f(x)+g(x)}{2})$. It is easy to see that h is a signed k'-majority if $s_A = s_B$, and a signed k-majority if s_A and s_B have Hamming distance $n - k$. In Lemma VI.3 below, we show that a signed k'-majority is a constant distance from any signed k-majority. Thus, Alice and Bob can solve $\text{GEQ}_{k'}$ by testing whether h is a signed k-majority.

Note that each time their tester queries $h(x)$, in order

to compute h they need to send $\Theta(\log k)$ bits to each other, since the range of f and g is of size $\Theta(k')$. Thus, similar to Lemma II.2, the communication complexity of this problem is bounded by $O(\log k')$ times the query complexity of testing. By Lemma II.3, we know that the communication complexity of $\text{GEQ}_{k'}$ with one-sided error is $\Omega(k')$. Thus, the query complexity of the tester must be $\Omega(k'/\log k') = \Omega(k/\log k)$. ∎

We complete the section by showing that when k' is much larger than k, signed k'-majorities are far from signed k-majorities. To prove this statement, we will use the Berry-Esseen theorem, a version of the Central Limit Theorem with error bounds (see e.g. [21]):

Theorem VI.2 (Berry-Esseen)**.** *Let $\ell(x) = c_1x_1 + \cdots + c_nx_n$ be a linear form over the random ± 1 bits x_i. Assume $|c_i| \leq \tau$ for all i and write $\sigma = \sqrt{\sum c_i^2}$. Write F for the c.d.f. of $\ell(x)/\sigma$; i.e., $F(t) = \Pr[\ell(x)/\sigma \leq t]$. Then for all $t \in \mathbb{R}$,*

$$|F(t) - \Phi(t)| \leq O(\tau/\sigma) \cdot \frac{1}{1 + |t|^3},$$

where Φ denotes the c.d.f. of X, a standard Gaussian random variable. In particular, if $A \subseteq \mathbb{R}$ is any interval then $|\Pr[\ell(x)/\sigma \in A] - \Pr[X \in A]| \leq C_1(\tau/\sigma)$, where C_1 is an absolute constant.

Lemma VI.3. *Fix a constant α. Then there exist absolute constants $k_0 \in \mathbb{N}$ and $\epsilon > 0$ (which only depend on α) such that for any $k \geq k_0$ and $k' = (1 + \alpha)k$, all signed k'-majorities are ϵ-far from signed k-majorities.*

Proof: Let f be a signed k-majority, and g be a signed k'-majority. It is easy to see that f and g have minimum distance when they have the same sign pattern on their common variables. So without loss of generality, assume $f(x) = \text{sgn}(x_1 + \cdots + x_k)$ and $g(x) = \text{sgn}(x_1 + \cdots + x_{k'})$ (in other words, f is a majority function on the first k' variables, and g is a majority function on the first k' variables). To simplify, we will write $S(x) = \sum_{i=1}^{k} x_i$ and $T(x) = \sum_{i=k+1}^{k'} x_i$. Thus, $f(x) = \text{sgn}(S(x))$ and $g(x) = \text{sgn}(S(x) + T(x))$.

For any positive real number t, we have

$$\begin{aligned}\Pr_x[f(x) \neq g(x)] &\geq \Pr_x[S(x) \in [0, t) \text{ and } T(x) < -t] \\ &= \Pr_x[S(x) \in [0, t)] \cdot \Pr_x[T(x) < -t]\end{aligned}$$

where the equality follows from the fact that S and T are functions on disjoint sets of variables.

Note that S is a linear form on k variables, so we can use the Berry-Esseen theorem on S with $\sigma = \sqrt{k}$ to get

$$\begin{aligned}\Pr_x[S(x) \in [0, t)] &\geq (\Phi(t/\sqrt{k}) - \Phi(0)) - C_1/\sqrt{k} \\ &\geq (\Phi(t/\sqrt{k}) - 1/2) - C_1/\sqrt{k} \quad (1)\end{aligned}$$

where C_1 is the constant from the Berry-Esseen theorem.

Similarly, T is a linear form on αk variables, so we can use the Berry-Esseen theorem on T with $\sigma = \sqrt{\alpha k}$ to get

$$\Pr_x[T(x) < -t] \geq \Phi(-t/\sqrt{\alpha k}) - C_1/\sqrt{\alpha k} \quad (2)$$

Setting t to be, say, $\sqrt{k}$, and then choosing k large enough insures that the quantities in both (1) and (2) are positive, and bigger than a constant which only depends on α. ∎

ACKNOWLEDGMENTS

We thank Amit Weinstein and the anonymous referees for insightful feedback on an earlier draft of this article. We also thank Sourav Chakraborty, David García-Soriano, and Arie Matsliah for sharing their manuscript [14] with us. In addition, E.B. wishes to thank Ryan O'Donnell for several helpful discussions.

REFERENCES

[1] Noga Alon and Eric Blais. Testing boolean function isomorphism. In *Proc. 14th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 394–405, 2010.

[2] Tugkan Batu, Ronitt Rubinfeld, and Patrick White. Fast approximate PCPs for multidimensional bin-packing problems. In *Proc. 3rd International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 245–256, 1999.

[3] Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Transitive-closure spanners of the hypercube and the hypergrid. Technical Report TR09-046, ECCC, 2009.

[4] Eric Blais. Improved bounds for testing juntas. In *Proc. 12th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 317–330, 2008.

[5] Eric Blais. Testing juntas nearly optimally. In *Proc. 41st Annual ACM Symposium on the Theory of Computing*, pages 151–158, 2009.

[6] Eric Blais and Daniel Kane. Testing linear functions. Manuscript, 2011.

[7] Eric Blais and Ryan O'Donnell. Lower bounds for testing function isomorphism. In *Proc. 25th Annual IEEE Conference on Computational Complexity*, pages 235–246, 2010.

[8] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *J. Comput. Syst. Sci.*, 47:549–595, 1993. Earlier version in STOC'90.

[9] Jop Briët, Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Monotonicity testing and shortest-path routing on the cube. In *Proc. 14th International Workshop on Randomization and Approximation Techniques in Computer Science*, 2010.

[10] Joshua Brody, Amit Chakrabarti, Oded Regev, Thomas Vidick, and Ronald de Wolf. Better Gap-Hamming lower bounds via better round elimination. In *Proc. 14th International Workshop on Randomization and Approximation Techniques in Computer Science*, 2010.

[11] Joshua Brody, Kevin Matulef, and Chenggang Wu. Lower bounds for testing computability by small-width branching programs. In *Proc. 8th Annual Theory and Applications of Models of Computation*, 2011.

[12] Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation. In *Proc. 30th Annual ACM Symposium on the Theory of Computing*, pages 63–68, 1998.

[13] Amit Chakrabarti and Oded Regev. An optimal lower bound on the communication complexity of Gap-Hamming-Distance. In *Proc. 43rd Annual ACM Symposium on the Theory of Computing*, 2011.

[14] Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Efficient sample extractors for juntas with applications. Manuscript, 2011.

[15] Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Nearly tight bounds for testing function isomorphism. In *Proc. 22nd Annual ACM-SIAM Symposium on Discrete Algorithms*, 2011.

[16] Hana Chockler and Dan Gutfreund. A lower bound for testing juntas. *Information Processing Letters*, 90(6):301–305, 2004.

[17] Ronald de Wolf. A brief introduction to fourier analysis on the boolean cube. *Theory of Computing, Graduate Surveys*, 1:1–20, 2008.

[18] Ilias Diakonikolas, Homin Lee, Kevin Matulef, Krzysztof Onak, Ronitt Rubinfeld, Rocco Servedio, and Andrew Wan. Testing for concise representations. In *Proc. 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 549–558, 2007.

[19] Yevgeniy Dodis, Oded Goldreich, Eric Lehman, Sofya Raskhodnikova, Dana Ron, and Alex Samorodnitsky. Improved testing algorithms for monotonicity. In *Proc. 3rd International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 97–108, 1999.

[20] Funda Ergun, Sampath Kannan, Ravi Kumar, Ronitt Rubenfeld, and Mahesh Viswanathan. Spot-checkers. *J. Comput. Syst. Sci.*, 60:717–751, 2000.

[21] W. Feller. *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons, 1968.

[22] Eldar Fischer, Guy Kindler, Dana Ron, Shmuel Safra, and Alex Samorodnitsky. Testing juntas. *J. Comput. Syst. Sci.*, 68:753–787, 2004.

[23] Eldar Fischer, Eric Lehman, Ilan Newman, Sofya Raskhodnikova, Ronitt Rubinfeld, and Alex Samorodnitsky. Monotonicity testing over general poset domains. In *Proc. 34th Annual ACM Symposium on the Theory of Computing*, pages 474–483, 2002.

[24] Peter Frankl and Vojtěch Rödl. Forbidden intersections. *Trans. Amer. Math. Soc.*, 300(1):259–286, 1987.

[25] Oded Goldreich. On testing computability by small width OBDDs. In *Proc. 14th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 574–587, 2010.

[26] Oded Goldreich, Shafi Goldwasser, Eric Lehman, Dana Ron, and Alex Samorodnitsky. Testing monotonicity. *Combinatorica*, 20(3):301–337, 2000.

[27] Johan Håstad and Avi Wigderson. The randomized communication complexity of set disjointness. *Theory of Computing*, pages 211–219, 2007.

[28] Piotr Indyk and David Woodruff. Tight lower bounds for the distinct elements problem. In *Proc. 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 283–289, 2003.

[29] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, Cambridge, 1997.

[30] Kevin Matulef, Ryan O'Donnell, Ronitt Rubinfeld, and Rocco Servedio. Testing {-1,1}-weight halfspaces. In *Proc. 13th International Workshop on Randomization and Approximation Techniques in Computer Science*, 2009.

[31] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. In *Proc. 27th Annual ACM Symposium on the Theory of Computing*, pages 103–111, 1995.

[32] Ilan Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 1991.

[33] Ryan O'Donnell. Some topics in analysis of boolean function. In *Proc. 40th Annual ACM Symposium on the Theory of Computing*, pages 569–578, 2008.

[34] Michal Parnas, Dana Ron, and Ronitt Rubinfeld. On testing convexity and submodularity. *SIAM J. Comput.*, 32(5):1158–1184, 2003.

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

[36] Dana Ron. Property testing: A learning theory perspective. *Foundations and Trends in Machine Learning*, 1(3):307–402, 2008.

[37] Dana Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5(2):73–205, 2009.

[38] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM J. Comput.*, 25:252–271, 1996.

[39] C. Seshadhri and Jan Vondrák. Is submodularity testable? In *Proc. 2nd Innovations in Computer Science*, 2011.